LF-VIDEOQA: EVALUATING LONG-VIDEO UNDERSTANDING WITH LONG-FORM THEORY-OF-MIND QUESTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

The hardest part of understanding a long video is not pinpointing any one moment but synthesizing many, e.g., answering a question that requires tracking what develops across hours of content and articulating how the pieces fit together. To date, most long-video QA datasets require only a short answer or a multiple-choice selection, oversimplifying the long-video understanding task to retrieval. We introduce LF-VIDEOQA, a dataset for long-form, open-ended question answering over long narrative videos built on theory-of-mind questions: a character's belief depends on everything they have seen or been told, so each question demands assembling and reasoning from evidence distributed across the video. The dataset contains more than 6,300 questions on 94 recent films; each requires reasoning over scenes spanning an average of 23 minutes and comes with a question-specific rubric for recall and precision scoring. The dataset is created entirely automatically by annotating screenplays with scene events and each character's first-order beliefs and meta-beliefs, then generating (question, answer, rubric) triples through LLM-based question generation and verification; human studies validate their quality. We evaluate four families of state-of-the-art open-source VLMs across three settings and find the task highly challenging: (i) even an oracle with access to the screenplay falls well short of complete answers; (ii) the best non-oracle model attains barely half of the recall level of the oracle; (iii) unlike on prior long-video benchmarks, captions and subtitles underperform the video itself, indicating that the questions need visual evidence; and (iv) recall is the predominant bottleneck, with answers omitting many required rubric claims.

1 INTRODUCTION

Understanding a long video means following how it develops as a whole, relating what happens now to what happened before (Wu & Krähenbühl, 2021). In a narrative video (i.e. Movies), events unfold over hours, people acquire new information and change their beliefs, and earlier interactions shape later decisions (Vicol et al., 2018; Papalampidi et al., 2019). The evidence needed to explain these developments is distributed across many scenes, none of which provides a complete answer on its own (Fung et al., 2023). Long-video understanding therefore requires synthesizing evidence from many moments into a coherent explanation.

Consider the question in Figure 1, asked near the end of *Deadpool & Wolverine* (2024): "At Wade's apartment, how has Logan's trust in Wade evolved since recruitment?" Logan's trust rests on a false belief. At minute 35, Wade (Deadpool) promises Logan (Wolverine) that the TVA (the agency that polices the film's timelines) can repair his ruined world. At minute 61, Wade admits he is not sure the TVA can fix anything, and then at minute 112, Logan sees the TVA decline to rewrite his past. A complete answer must trace this arc across 83 minutes of the film, and no single scene contains both the belief and its correction. Long-form answers make each of these steps assessable by exposing which connections a model draws, which claims it supports, and which parts of the explanation it omits, since a long answer can be decomposed into claims that are checked one by one (Min et al., 2023; Song et al., 2024b). These, theory-of-mind (ToM) question, indeed can capture the true nature of long-output and long-input.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

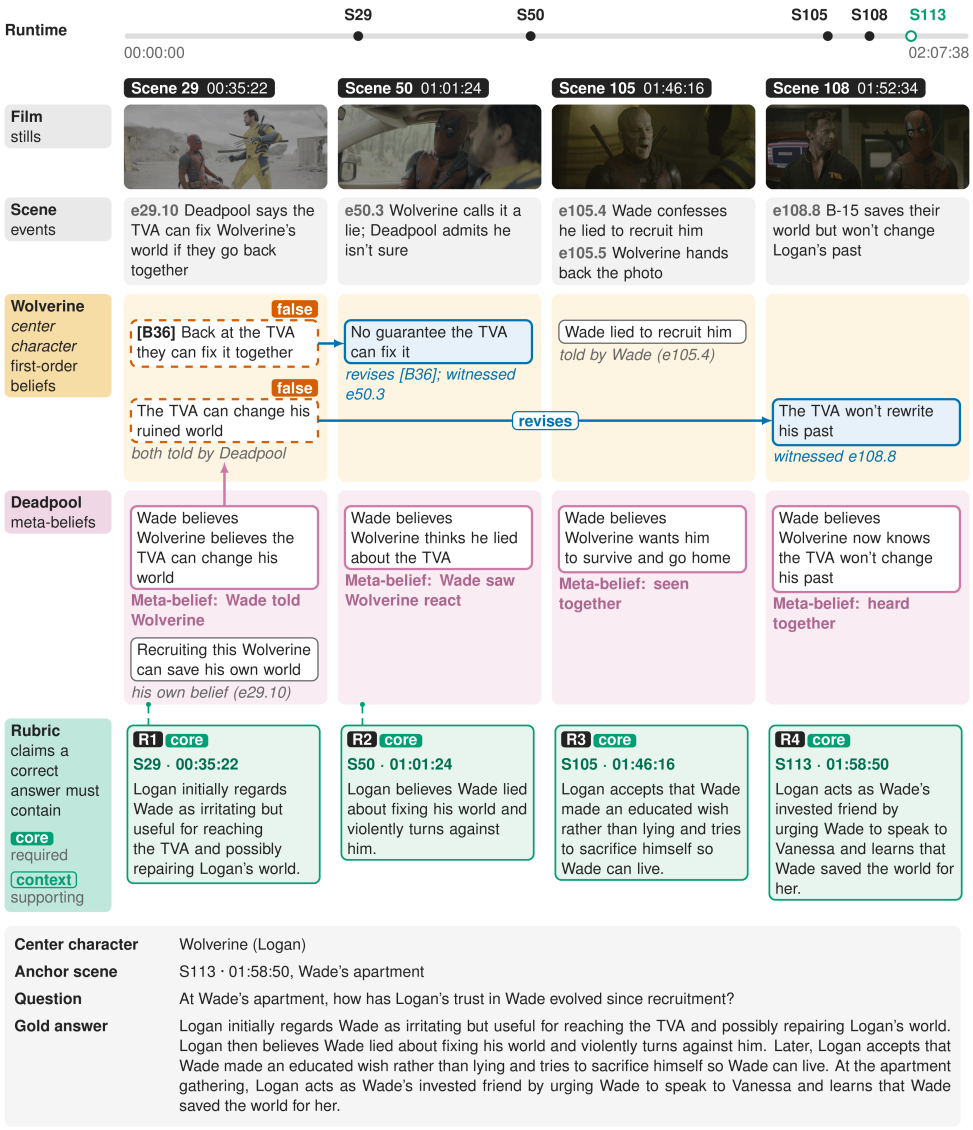


Figure 1: **Dataset Construction on an Example.** This figure shows (question, answer, rubric) from *Deadpool & Wolverine* (2024). *Preprocessing:* screenplay scenes are time-stamped against the subtitles and decomposed into scene events. *Character belief annotation:* scene by scene, we extract and verify each character's first-order beliefs and meta-beliefs. *Question generation and verification:* for a center character (Wolverine) and an anchor (scene 113), we generate the triple and verify every rubric claim against the screenplay.

Existing long-video question-answering benchmarks predominantly use multiple-choice or short-answer formats (Mangalam et al., 2023; Fu et al., 2024; Wu et al., 2024; Wang et al., 2024a). Although they provide long inputs, many of their questions can be answered by retrieving a few relevant frames or short segments (Mangalam et al., 2023), and the multiple-choice format further allows answering by elimination. Moreover, captions, subtitles, or even the question alone can support strong performance on many long-video benchmarks (Zhang et al., 2024), raising questions about the degree to which they depend on visual information.

We introduce LF-VIDEOQA, a dataset for long-form, open-ended ToM question answering over long narrative videos, with more than 6,300 questions on 94 recent films. Both the input and the output are long: answering each question requires reasoning over an average of 4.2 scenes spanning 23 minutes, and reference answers average 61 words. The questions follow a film's cast rather

than its protagonist, asking about a median of 15.5 characters per film. Our construction pipeline is fully automatic (Figure 1): it annotates each film's screenplay, long used as a structured text view of films (Papalampidi et al., 2019; 2020; Tian et al., 2026), with scene events and each character's first-order beliefs and meta-beliefs (beliefs about other characters' beliefs), then generates (question, answer, rubric) triples through multiple rounds of LLM-based question generation and verification. Each rubric lists the claims a correct answer should make, verified against the screenplay, and supports automatic scoring of recall and precision beyond surface similarity to a reference answer. Human studies validate the quality and interpretability of the resulting questions, answers, and rubrics.

We evaluate four families of state-of-the-art open-source vision-language models and find substantial room for improvement. Even an oracle with access to the screenplay falls well short of producing complete answers, while the best non-oracle model achieves barely half the oracle's recall. Video input also outperforms text-based representations such as captions and subtitles, suggesting that these representations omit visual evidence needed to answer the questions. Across models and settings, recall is the predominant bottleneck: model answers leave out many of the rubric claims required for a complete response. Together, these findings reveal a persistent gap between identifying relevant information and incorporating that information into a sufficiently complete explanation.

Our contributions are:

- **LF-VIDEOQA**, a dataset for long-form, open-ended QA over long narrative videos, with questions requiring evidence from distant scenes and question-specific evaluation rubrics.
- **A fully automatic construction pipeline** that uses screenplay-derived events, beliefs, and meta-beliefs to generate and verify question–answer–rubric triples, with quality validated through human studies.
- **An evaluation of four open-source VLM families** showing substantial difficulty producing complete answers, an advantage for video over text-based representations, and missing required claims as the predominant performance bottleneck.

2 RELATED WORK

Long inputs, short outputs. Video QA benchmarks have grown from clips of seconds to videos of an hour or more, but the answers they require have mostly stayed short (Table 5 in Appendix A). Most long-video benchmarks are multiple-choice (Mangalam et al., 2023; Fu et al., 2024; Wu et al., 2024; Wang et al., 2024a; Chandrasegaran et al., 2024), as are most film and TV benchmarks (Tapaswi et al., 2016; Lei et al., 2018; Zhang et al., 2023; Rawal et al., 2024), and many open-ended benchmarks accept a phrase: answers average 1.9 words in ActivityNet-QA (Yu et al., 2019), 2.6 in NExT-QA (Xiao et al., 2021), and 2.3 in MovieChat-1K (Song et al., 2024a). Longer outputs appear in two other settings, neither of which matches ours. Video summarization produces multi-sentence outputs (Zhou et al., 2025b; Papalampidi & Lapata, 2023; Mahon & Lapata, 2024), but they describe the video rather than answer a question about it. Recent benchmarks pose open-ended questions about long narrative videos (Fung et al., 2023; Ataallah et al., 2025; Jain et al., 2026), but their reference answers are short (19–27 tokens in DeepMovieQA). LF-VIDEOQA asks for question-specific explanations averaging 61 words, decomposed into verified claims, so we measure which required claims an answer states, omits, or contradicts rather than assigning it a single score.

Theory-of-mind questions. Theory of mind is classically tested with false-belief questions answered in a word, such as "Where will Sally look for her marble?". However, we expand that in out work to include other variaents of belief states. We FANToM adds free-form questions about what participants who missed part of a conversation believe (Kim et al., 2023), and CharToM-QA grades answers about novel characters against several key points (Zhou et al., 2025a). Video ToM benchmarks ask about short clips (Sap et al., 2019), household activities (Jin et al., 2024), or short films (Villa-Cueva et al., 2025) with multiple-choice answers; MMToM-QA, for instance, asks which of two beliefs a person more likely holds. MovieGraph-ToM asks about feature films but gives models selected keyframes and script segments rather than the film (Wei et al., 2026). A question such as "how has Logan's trust in Wade evolved since recruitment?" needs evidence spread over 83 minutes of the film and an answer with several claims, the combination LF-VIDEOQA targets.

Video models and agents. LLoVi (Zhang et al., 2024) showed that a simple pipeline is a strong long-video baseline: a captioner describes short clips, and an LLM answers from the concatenated captions, improving the previous best result on EgoSchema by 18 points. Agents instead search the video with tools, retrieving and inspecting a few relevant moments before answering (Wang et al., 2024b). Both designs reduce a video to captions or to a few selected moments. Our text-based baselines follow LLoVi's caption-then-reason design, and we also evaluate an agentic baseline, to test whether these reductions keep the evidence a long-form answer about a character's beliefs needs.

3 Dataset and Benchmark Design

We want questions whose input and output are both long: answering them needs evidence distributed across the video, and the answer takes more than a short phrase. Theory-of-mind (ToM) questions have both properties. Following Jin et al. (2024), we treat a character's belief at a given moment as the product of what they have observed up to that point, which may differ from what is actually true (a false belief). Answering such a question also takes a set of claims rather than a single phrase. Long-form answers are hard to score against a single reference answer (Xu et al., 2023). We therefore follow claim-level evaluation of long-form text, which decomposes an answer into claims that are checked one by one (Min et al., 2023; Song et al., 2024b), but change what the claims are checked against. We pair each question with a *rubric*: a list of claims a correct answer should make, each describing a single event or state and grounded in the screenplay scenes that support it. Given a model's answer, we check which of the required claims it states (recall) and, of the rubric claims it commits to, the share it states correctly rather than contradicts (precision; Section 3.3). The rest of this section describes how we build these *(question, answer, rubric)* triples. Figure 1 follows one of them through the construction pipeline, and Table 1 summarizes the eleven question types we generate (Section 3.4).

Question type	n	Core claims	Context claims	Question tokens	Answer tokens	Claim tokens	Cited scenes	Span (min)	Anchor start	Anchor end
■ Q1 Knowledge split	206	3.0	1.3	23	77	14	3.1	7		
■ Q2 Realization	857	1.5	1.8	20	75	17	4.1	24		
■ Q3 False belief	325	1.4	2.1	20	73	14	3.4	12		
■ Q4 Knowledge source	1,463	1.6	1.2	20	69	16	3.0	5		
■ Q5 First-order belief	128	1.3	1.3	21	65	16	3.2	9		
■ Q6 Second-order belief	112	1.2	1.2	23	61	17	3.0	5		
■ Q7 Accepting a claim	1,151	2.0	1.3	22	73	16	3.0	4		
■ Q8 Order of discovery	535	4.1	1.1	23	85	16	5.7	52		
■ N1 Epistemic arc	589	4.7	1.9	23	94	16	7.9	68		
■ N2 Trust evolution	475	2.5	1.6	21	68	15	6.4	60		
■ N3 Causal chain	486	4.0	1.2	22	80	15	3.7	15		
All questions	6,327	2.4	1.4	21	75	16	4.2	23		

Table 1: Statistics per question type. ■ Belief-state types ask about one moment; ■ arc types ask about the whole film. All columns except n are averages per question; span is the runtime between the earliest and latest cited scene; the last column shows where questions are anchored.

3.1 Data Collection and Preprocessing

For 94 films released in 2023 or later, we collect *screenplays*, *subtitles*, and character rosters from public sources (ScriptSlug, OpenSubtitles, and TMDB, respectively).[1] We restrict to recent releases to reduce the chance that the films, or discussions of them, appear in the training data of the vision–language models we evaluate.

To generate our *(question, answer, rubric)* triples, we use the screenplay as the ground-truth text representation of each film. We use a language model (Qwen3.5-27B) to split each screenplay into *scenes* with speaker-attributed dialogue and stage directions, summarize every scene, and resolve

[1] `https://www.scriptslug.com`, `https://www.opensubtitles.com`, `https://www.themoviedb.org`

coreferent mentions of characters. Finally, we time-stamp each scene by aligning its dialogue with the subtitles, which places every scene, and every claim later built from it, on the film's timeline. Appendix C details each preprocessing step with the prompts used.

Character Belief Annotation. From each parsed screenplay, we build a per-character *belief record* with the same model, working through the film scene by scene. We first decompose every scene into atomic *scene events*, each listing its participants and witnesses. For *first-order beliefs*, we go through the scenes in order for each character, giving the model the scene, its events, and the character's beliefs so far. It determines what the character perceived firsthand, what they were told and whether they believed it, what they inferred, which of their beliefs are false, and which earlier beliefs they revised; the result is carried into the next scene. For example, in Figure 1, Wolverine accepts two false promises from Deadpool in scene 29 and revises them in scenes 50 and 108, when what he witnesses contradicts them. For *meta-beliefs*, we again go scene by scene and record what each character A comes to believe about the beliefs of another character B, along with how that belief formed. Details and prompts are in Appendix D.

3.2 Question Generation and Verification

Writing questions from context-sets. Every question is built around two choices. The *center character* is the character whose mind the question asks about. The *anchor* is the scene that fixes the moment the question refers to. The question points to it in a few words by a place, an occasion, or an object, never by what happens there (e.g., "At Wade's apartment"), so a model must first locate that moment in the video and then reconstruct what the center character knew by then. Each question also has one of eleven *types*, which fixes the kind of epistemic state it asks about, such as how the center character comes to know a fact or how their view of another character changes, and the parts of the belief record it draws on (Section 3.4).

We never give the question writer the whole film. Instead, each question is written from a *context-set* for one center character, anchor, and type. We discard context-sets based on rules, then we sample the remaining context-sets. From each context-set, the question writer (gpt-5.6-sol) produces a candidate question and an outline of claims or declines to write one. Appendix F provides the prompts and discard rules.

Rubrics and question verification. Because the belief record is extracted automatically, it can be wrong, and so can the claims drafted from it. We therefore verify every claim against the screenplay itself. The verifier (gpt-5.6-sol) labels each claim as supported, unsupported, or contradicted by the relevant parts of the screenplay. We remove the claims the verifier does not support and rewrite those that are only partly supported. If the surviving claims no longer answer the question, the verifier rewrites the question so that they answer it completely; about one in ten released questions was revised this way, and questions with too few surviving core claims are discarded. The surviving claims form the question's rubric. Each claim is either *core*, a claim the question asks for and a complete answer must make, or *context*, a premise an answer may omit but must not contradict. We then write the reference answer from the rubric as a short paragraph.

3.3 Evaluation metrics

For a question q, let $\mathcal{K}_q$ be its rubric and $\mathcal{C}_q \subseteq \mathcal{K}_q$ its core claims. Given a model answer a_q, an LLM judge (gpt-5.6-luna) labels every rubric claim as *asserted*, *contradicted*, or *omitted* by the answer, judging only against the answer's text (Appendix F.2). Let $\mathcal{A}_q$ and $\mathcal{X}_q$ be the rubric claims the answer asserts and contradicts. We compute:

$$\mathrm{R}_q = \frac{|\mathcal{A}_q \cap \mathcal{C}_q|}{|\mathcal{C}_q|}, \qquad \mathrm{P}_q = \frac{|\mathcal{A}_q|}{|\mathcal{A}_q| + |\mathcal{X}_q|}, \qquad \mathrm{F1}_q = \frac{2\,\mathrm{P}_q\,\mathrm{R}_q}{\mathrm{P}_q + \mathrm{R}_q}, \tag{1}$$

We report each measure averaged over questions. We also report the contradiction rate $|\mathcal{X}_q|/|\mathcal{K}_q|$.

3.4 Dataset Statistics

We define eleven question types by the parts of the belief record they draw on (Table 1). Eight ask about the center character at one moment: how several characters' knowledge of the same situation differs (Q1), when their understanding changes (Q2), what they take to be the case right before acting on a false belief (Q3), how they come to know a fact (Q4), what they take another character to believe (Q5) or to believe about a third (Q6), what they make of a claim they are told (Q7), and the order in which they learn a set of facts (Q8). The other three ask about the center character across the film: what they come to know and how (N1), how their view of another character changes (N2), and the chain of events that leads them to a belief (N3).

	Overall	Per film
Films	94	–
Runtime	182 h	112 min
Scenes	11,712	118
Questions	6,327	60
Verified claims	24,496	250
Unique Characters	1,625	15.5
Tokens		
– subtitles	1.35M	14K
– screenplay	2.96M	31K

Table 2: **Dataset Size**

Character coverage. Importantly, the questions follow a film's entire cast and not just its protagonist (Table 2). A film has a median of 15.5 center characters, and its most-questioned one accounts for only about a fifth of its questions. Meaning a model that tracks only the main character therefore answers a minority of the dataset.

Scene coverage. Every question cites at least two scenes and on average 4.2, spanning a mean of 23 minutes of runtime (Table 1); for arc questions the median span is 44 minutes. Together, the questions of a film cite a median of 71% of its scenes, which cover 72% of its runtime, and their anchors are spread almost uniformly over the film, with each fifth of the runtime holding between 16% and 23% of the questions. No part of a film can therefore be skipped. Reference answers average 61 words and contain 2.4 core and 1.4 context claims.

3.5 Human Study – Dataset Validation

Two annotators independently assessed each of 150 questions, their reference answers, and 494 associated claims. Both annotators recommended keeping 83.3% of questions and 85.3% of reference answers; both components received unanimous keep decisions for 76.7% of questions. At the claim level, both annotators judged 80.4% factually supported and 75.5% supported by sufficient linked evidence. Exact agreement ranged from 80.6% to 96.6% across criteria. These results document predominantly favorable assessments of the inspected subset, but limited chance-adjusted agreement constrains conclusions about the reproducibility of those assessments.

4 Models and Experimental Settings

We construct a diverse set of baselines to isolate the challenges of perception, long-context understanding and evidence acquisition. Unless otherwise stated, all models are evaluated zero-shot.

4.1 Text-based Baselines

Our first set of baselines removes the complexity of perception by converting each movie into a time-ordered textual representation.

Screenplay (oracle). This oracle receives the movie screenplay – the ground truth that was used to generate the question in the first place. While impractical, it allows us to see if models can extract relevant information from the screenplay without the carefully crafted pipeline used during question generation.

Subtitles. This baseline receives the movie's time-aligned subtitle track (and the question). It measures how well the task can be solved from dialogue alone without any visual information.

Video captions. We divide the movie into 10-second chunks and caption each using a video-language model (VLM). We annotate each video with character names to provide character identity to the VLM. The resulting captions are concatenated and provided to the model. This baseline

removes dialogue and therefore measures the extent to which questions can be answered from visual information alone. More information can be found in Appendix I.

Video captions + subtitles. This baseline receives the 10-second video captions interleaved with the subtitles corresponding to the same temporal interval. This condition provides both visual information and dialogue while preserving their temporal organization.

4.2 Video-Based Baselines

The previous baselines may discard visual information before the answering model observes it. To see if question-dependent visual processing is necessary, we consider two video understanding baselines. **(1) Video without audio:** The model receives uniformly sampled visual input (at 0.2fps) from the movie together with the question, but no audio. **(2) Video without audio + subtitles:** This baseline receives the same visual input together with the complete time-aligned subtitle track, which provides dialogue. This condition evaluates whether the model can answer narrative questions from the visual stream alone. We run this setting with are Qwen3.6 and Gemma4 as they are some of the few models that can handle video of that length.

4.3 Agentic approaches

The preceding baselines receive a fixed representation of the movie in a single model invocation. We also evaluate an agentic baseline that gets tools to iteratively search the movie, inspect selected intervals, and accumulate evidence before producing its final answer. We use Qwen3.6 35B/A3B as the agentic orchestrator. At each step, the model may call one of the following tools:

- `subtitle_search`: retrieves subtitle segments relevant to a query.
- `find_clips`: returns the temporal spans of clips that best match a query.
- `vqa`: answers a free-form question about a selected video interval.
- `browse_clip`: provides a coarse-grained summary of what happens in a video clip.
- `character_presence`: returns all characters appearing in a specified clip.

In addition to the **regular** agent that uses the above tools, we also consider a **fast** agent that uses precomputed 10-second video captions as a representation to do approximate search. At the other extreme, we also evaluate an **Oracle** agent that gets access to the screenplay.

5 Results and Discussion

Table 3 reports performance across text-based, video-based, and agentic settings. We report four lexical measures—RL, ClmRL, Cov@.5, and Hedge. Additionally, we also report recall, precision, F1, and contradiction rate from the LLM-based evaluator. Higher values are better for all metrics except Hedge and contradiction.

Screenplay-based Oracles Are Still Not Perfect As one might expect, oracles using the screenplay setting produces the strongest results by a substantial margin: Qwen3.8 27B obtains the highest LLM-judge F1 score of 0.544, with a recall of 0.509 and precision of 0.703. This is not surprising, since screenplays include not just dialogue but also actions, settings and other narratively important information. Nevertheless, even the best screenplay-based model attains only 0.509 recall, leaving nearly half of the evaluated answer content unrecovered. This suggests that recovering and synthesizing evidence from long context is still a challenge.

Subtitles Alone Are Insufficient for Narrative Understanding Replacing the screenplay with subtitles causes a large performance decline: for example, for Qwen3.8, LLM-judge F1 falls from 0.544 with the screenplay to 0.175 with subtitles. Claim coverage also decreases from 0.348 to 0.252 for Qwen3.6 and from 0.325 to 0.257 for Qwen3.8.

This gap shows that access to the complete dialogue does not provide a complete representation of the narrative. Many answers depend on actions, reactions, settings, and causal connections that are not explicitly verbalized.

Source	Model	Lexical				LLM judge			
		RL	ClmRL	Cov@.5	Hedge↓	Rec	Prec	F1	Contra↓
Gold answer		1.000	.686	.975	.007	–	–	–	–
TEXT-BASED									
Screenplay	Qwen3.6 35B/A3B	.160	**.248**	**.348**	.343	.472	.667	.505	**.128**
	Qwen3.8 27B⋆	**.170**	.243	.325	.216	**.509**	**.703**	**.544**	.103
	Gemma4 26B/A4B	.164	.220	.287	**.519**	.260	.464	.290	.100
	Keye-VL-1.5 8B	.140	.189	.193	.008	.143	.256	.162	.060
	InternVL3.5 8B	.142	.198	.165	.211	.072	.120	.081	.016
Subtitles	Qwen3.6 35B/A3B	.141	**.209**	.252	.770	.149	.261	.165	.145
	Qwen3.8 27B⋆	**.150**	.206	**.257**	.573	**.157**	**.275**	**.175**	**.163**
	Gemma4 26B/A4B	.139	.182	.206	**.937**	.055	.128	.064	.061
	Keye-VL-1.5 8B	.138	.183	.160	.009	.087	.166	.100	.080
	InternVL3.5 8B	.138	.188	.146	.271	.005	.011	.006	.005
Video Captioning	Qwen3.6 35B/A3B	.134	.188	**.211**	.891	.013	.041	.016	.078
	Qwen3.8 27B	.126	.180	.194	.544	.003	.016	.004	.024
	Gemma4 26B/A4B	.123	.167	.183	**.930**	.001	.007	.002	.017
	Keye-VL-1.5 8B⋆	**.144**	**.212**	.196	.120	**.062**	**.130**	**.071**	**.097**
	InternVL3.5 8B	.137	.186	.147	.235	.001	.002	.001	.001
Video Captioning + Subtitles	Qwen3.6 35B/A3B⋆	**.155**	**.229**	**.283**	.577	**.222**	**.368**	**.244**	**.169**
	Qwen3.8 27B	.133	.197	.224	.307	.099	.182	.112	.056
	Gemma4 26B/A4B	.132	.179	.177	**.641**	.031	.065	.036	.021
	Keye-VL-1.5 8B	.148	.217	.205	.102	.076	.151	.087	.114
	InternVL3.5 8B	.137	.187	.146	.234	.004	.008	.004	.002
VIDEO-BASED									
Video w/o Audio	Qwen3.6 35B/A3B⋆	**.145**	**.191**	.101	.012	**.086**	**.172**	**.099**	**.099**
	Gemma4 26B/A4B	.112	.144	**.102**	**.756**	.002	.009	.003	.062
Video w/o Audio + Subtitles	Qwen3.6 35B/A3B⋆	**.167**	**.242**	**.257**	.118	**.279**	**.452**	**.304**	.167
	Gemma4 26B/A4B	.146	.193	.159	**.692**	.062	.135	.071	**.226**
AGENTIC									
Qwen3.6 35B/A3B	Oracle⋆	**.202**	**.290**	**.194**	.124	**.302**	**.484**	**.332**	.093
	Fast	.190	.276	.163	.171	.208	.352	.229	.113
	Regular	.186	.270	.150	**.197**	.181	.312	.201	**.115**

Table 3: **Results on LF-VIDEOQA.** Rows are answerer models; in the video-captioning answerer also write the captions. Rec, Prec, F1, and Contra follow Section 3.3. Blue: five best values across all runs; red: five worst values of Hedge and Contra. **Bold**: highest value within a source.

Visual Information Is Most Useful When Combined With Language Conversely, automatically generated video captions perform poorly when used as the only movie representation. The strongest caption-only system, Keye-VL-1.5, obtains an LLM-judge F1 of 0.071. These results suggest that local visual descriptions alone do not preserve enough information to reconstruct the broader narrative.

Combining video captions with subtitles substantially improves Qwen3.6. Its F1 increases from 0.016 with captions alone and 0.165 with subtitles alone to 0.244 when the two are combined. Recall rises to 0.222 and precision to 0.368. This improvement shows that the two representations provide complementary information: subtitles capture spoken content, while captions introduce visible actions and settings.

The gain is not universal, however. Qwen3.8 obtains an F1 of 0.112 with video captions and subtitles, below its subtitle-only score of 0.175. Thus, providing additional information does not guarantee that a model will use it effectively. The model must still identify relevant observations and integrate them with the dialogue without being distracted by noisy or redundant descriptions.

Direct video input follows a similar pattern, where combining audio with video yields much better performance than video alone (0.304 F1 with subtitles vs 0.099 without for Qwen3.6). The direct-video result is stronger than the corresponding caption-based result for Qwen3.6: video with subtitles obtains 0.304 F1, compared with 0.244 for video captions with subtitles. This suggests that converting the video into textual descriptions discards information that the multimodal model can recover from the original visual input. At the same time, all these accuracies lag significantly behind the screenplay-based oracles, suggesting that models struggle to extract all relevant information.

Oracle Evidence Improves Agentic Performance The agentic experiments evaluate whether iterative evidence search can overcome the limitations of fixed-context inputs. Using Qwen3.6 as

Context	Captioner / tier	Belief & knowledge Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Narrative N1	N2	N3	All
Screenplay	–	**.64** ± .05	**.39** ± .03	**.61** ± .04	**.60** ± .03	.31 ± .10	.20 ± .07	**.61** ± .02	**.33** ± .03	**.35** ± .03	**.49** ± .04	**.55** ± .03	**.50** ± .01
Subtitles	–	.22 ± .04	.10 ± .02	.27 ± .04	.18 ± .02	.09 ± .05	.09 ± .05	.20 ± .03	.13 ± .02	.16 ± .02	.16 ± .03	.14 ± .03	.16 ± .01
Video Captioning	Qwen3.6 35B/A3B	.02 ± .01	.01 ± .01	.00 ± .01	.01 ± .01	.00 ± .00	.00 ± .00	.01 ± .00	.02 ± .01	.04 ± .01	.03 ± .01	.03 ± .01	.02 ± .00
	Qwen3.8 27B	.01 ± .01	.01 ± .01	.01 ± .01	.01 ± .01	.00 ± .00	.00 ± .00	.00 ± .00	.01 ± .01	.02 ± .01	.02 ± .01	.02 ± .01	.01 ± .00
	Gemma4 26B/A4B	.01 ± .01	.00 ± .01	.00 ± .00	.01 ± .01	.00 ± .00	.00 ± .00	.00 ± .00	.01 ± .01	.01 ± .01	.02 ± .01	.01 ± .01	.01 ± .00
	Keye-VL-1.5 8B	.00 ± .00	.00 ± .00	.00 ± .00	.01 ± .00	.00 ± .00	.00 ± .00	.00 ± .00	.01 ± .00	.01 ± .00	.01 ± .01	.01 ± .01	.00 ± .00
	InternVL3.5 8B	.00 ± .00	.00 ± .00	.00 ± .00	.01 ± .01	.00 ± .00	.00 ± .00	.00 ± .00	.01 ± .01	.01 ± .01	.01 ± .01	.01 ± .00	.00 ± .00
Video Captioning + Subtitles	Qwen3.6 35B/A3B	.33 ± .05	.16 ± .02	.30 ± .05	.26 ± .03	.21 ± .07	.09 ± .05	.32 ± .02	.19 ± .03	.22 ± .03	.26 ± .04	.21 ± .03	.24 ± .02
	Qwen3.8 27B	.28 ± .05	.17 ± .03	.31 ± .05	.24 ± .03	.21 ± .06	.14 ± .06	.30 ± .03	.17 ± .02	.22 ± .03	.21 ± .04	.20 ± .03	.23 ± .02
	Gemma4 26B/A4B	.28 ± .05	.14 ± .02	.30 ± .04	.23 ± .03	.15 ± .06	.13 ± .06	.28 ± .03	.16 ± .03	.20 ± .03	.22 ± .03	.20 ± .03	.22 ± .01
	Keye-VL-1.5 8B	.18 ± .04	.09 ± .02	.21 ± .05	.12 ± .02	.11 ± .06	.13 ± .07	.18 ± .03	.12 ± .02	.13 ± .02	.11 ± .03	.09 ± .03	.13 ± .01
	InternVL3.5 8B	.28 ± .05	.14 ± .02	.29 ± .05	.20 ± .03	.17 ± .07	.15 ± .06	.27 ± .03	.16 ± .03	.19 ± .03	.17 ± .03	.15 ± .02	.20 ± .02
Video w/o Audio	–	.10 ± .03	.06 ± .02	.16 ± .04	.08 ± .02	.12 ± .06	.10 ± .07	.17 ± .02	.05 ± .02	.08 ± .02	.14 ± .04	.05 ± .02	.10 ± .01
Video w/o Audio + Subtitles	–	.37 ± .06	.20 ± .04	.44 ± .06	.33 ± .03	**.35** ± .09	**.28** ± .10	.39 ± .03	.19 ± .03	.25 ± .03	.33 ± .05	.20 ± .03	.30 ± .02
Qwen3.6 35B/A3B	Oracle	.34 ± .05	.28 ± .03	.42 ± .05	.46 ± .03	.29 ± .09	.22 ± .08	.39 ± .03	.18 ± .02	.17 ± .02	.29 ± .04	.26 ± .03	.33 ± .02
	Fast	.22 ± .04	.16 ± .03	.35 ± .05	.29 ± .03	.20 ± .07	.16 ± .07	.31 ± .02	.13 ± .02	.14 ± .02	.19 ± .03	.16 ± .03	.23 ± .02
	Regular	.21 ± .04	.13 ± .02	.29 ± .05	.26 ± .03	.19 ± .07	.13 ± .06	.28 ± .03	.11 ± .02	.12 ± .02	.16 ± .03	.13 ± .03	.20 ± .01

Table 4: **Qwen3.6 across question types.** The answerer is always Qwen3.6; rows vary the source and the captioner. Cells are F1 (w/ 95% bootstrap interval). Blue: five highest per question type, darker for higher; **bold**: highest.

the orchestrator, the Regular agent obtains 0.181 recall, 0.312 precision, and 0.201 F1. The Fast configuration improves these scores to 0.208 recall, 0.352 precision, and 0.229 F1. This is still substantially worse than simply providing the video and subtitles as input to Qwen3.6 (0.304 F1).

The Oracle agent performs substantially better, reaching 0.302 recall, 0.484 precision, and 0.332 F1. This result confirms that evidence localization is an important bottleneck: the agent answers substantially more of the question when it can retrieve higher-quality evidence.

However, Oracle remains well below the screenplay-based Qwen3.6 result, obtaining 0.332 F1 compared with 0.505. Oracle retrieval therefore does not eliminate the difficulty of the task. Even when the agent receives improved evidence, it must still determine which observations matter, connect events across the movie, and synthesize them into a complete answer.

6 Conclusion

We introduce a benchmark for evaluating whether models can understand feature-length narrative videos and produce detailed, open-ended answers. Unlike settings centered on short clips, localized evidence, or brief responses, our task requires models to track characters and events across an entire movie, reason about implicit beliefs, motivations, and relationships, and synthesize multiple pieces of evidence into a coherent explanation. To evaluate these responses beyond surface-level similarity, the dataset provides question-specific rubrics that measure both the completeness and faithfulness of generated answers.

Our experiments reveal limitations at multiple stages of current long-video systems. Models perform best when movies are represented by narratively structured screenplays, while performance declines substantially with subtitles, generated video captions, and direct video input. Combining visual information with subtitles recovers some missing information, but does not close this representation gap. Agentic evidence search also helps: oracle evidence substantially improves performance over regular retrieval, demonstrating that evidence localization is an important bottleneck. However, the remaining gap under oracle access shows that retrieval alone is insufficient. Models continue to struggle to integrate temporally distributed evidence and express all components of a long-form answer. These findings suggest that feature-length video understanding is not solely a context-length problem. Progress will require better preservation of narratively important visual information, more reliable evidence localization, stronger character and event tracking, and models capable of synthesizing retrieved observations into complete, grounded explanations. We hope this benchmark provides a foundation for studying these challenges and developing systems that can understand extended multimodal narratives rather than merely recognize isolated moments.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic data sets. The questions, reference answers, and rubrics of LF-VIDEOQA are generated with OpenAI models: gpt-5.6-sol writes them and verifies each claim against the screenplay, and gpt-5.6-luna checks the reference answers and serves as the LLM judge (Sections 3.2 and 3.3). Qwen3.5-27B parses the screenplays into scenes and builds the belief records (Section 3.1). All prompts are given in Appendices F.2. We used Claude Code (Anthropic) as a coding assistant to implement the construction pipeline and the experiments.

We have **not** used generative AI tools to propose or design pipelines, experiments and analysis: the research questions, the dataset design, the experiments, and the conclusions are our own. Additionally, we used generative AI tools (Claude) to create and edit software code, to build the prompt and example tables in the appendix, and to edit the text of the paper for readability. We have reviewed all AI-assisted work. Two annotators independently reviewed 150 generated questions together with their reference answers and claims (Section 3.5). We reviewed and tested all AI-assisted code, checked the generated tables against their source files, and checked every edit to the text.

ETHICS STATEMENT

LF-VIDEOQA is a question-answering dataset on films. We do not release or redistribute the videos, screenplays, subtitles, or cast photos. Instead, we provide scripts that let users download the screenplays, subtitles, and cast photos from the public sources we collected them from (Section 3.1); the scripts do not cover the videos, and we ask users to obtain them legally. All of this work is for non-commercial research use only.

REFERENCES

Kirolos Ataallah, Eslam Mohamed Bakr, Mahmoud Ahmed, Chenhui Gou, Khushbu Pahwa, Jian Ding, and Mohamed Elhoseiny. InfiniBench: A benchmark for large multi-modal models in long-form movies and TV shows. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 19485–19512, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.984. URL https://aclanthology.org/2025.emnlp-main.984/.

Keshigeyan Chandrasegaran, Agrim Gupta, Lea M. Hadzic, Taran Kota, Jimming He, Cristobal Eyzaguirre, Zane Durante, Manling Li, Jiajun Wu, and Li Fei-Fei. Hourvideo: 1-hour video-language understanding. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

Chaoyou Fu, Yuhan Dai, Yondong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024.

Yi Fung, Han Wang, Tong Wang, Ali Kebarighotbi, Mohit Bansal, Heng Ji, and Prem Natarajan. DeepMaven: Deep question answering on long-distance movie/TV show videos with multimedia knowledge extraction and synthesis. In Andreas Vlachos and Isabelle Augenstein (eds.), *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 3041–3051, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.221. URL https://aclanthology.org/2023.eacl-main.221/.

Rahul Jain, Keval Doshi, Burak Uzkent, and Garin Kessler. Narrative aligned long form video question answering, 2026. URL https://arxiv.org/abs/2603.19481.

Chuanyang Jin, Yutong Wu, Jing Cao, Jiannan Xiang, Yen-Ling Kuo, Zhiting Hu, Tomer Ullman, Antonio Torralba, Joshua Tenenbaum, and Tianmin Shu. MMToM-QA: Multimodal theory of mind question answering. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1:*

Long Papers), pp. 16077–16102, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.851. URL https://aclanthology.org/2024.acl-long.851/.

Hyunwoo Kim, Melanie Sclar, Xuhui Zhou, Ronan Bras, Gunhee Kim, Yejin Choi, and Maarten Sap. FANToM: A benchmark for stress-testing machine theory of mind in interactions. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14397–14413, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.890. URL https://aclanthology.org/2023.emnlp-main.890/.

Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. TVQA: Localized, compositional video question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 1369–1379, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1167. URL https://aclanthology.org/D18-1167/.

Louis Mahon and Mirella Lapata. A modular approach for multimodal summarization of TV shows. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8272–8291, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.450. URL https://aclanthology.org/2024.acl-long.450/.

Karttikeya Mangalam, Raiymbek Akshkulakov, and Jitendra Malik. Egoschema: a diagnostic benchmark for very long-form video language understanding. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12076–12100, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.741. URL https://aclanthology.org/2023.emnlp-main.741/.

Pinelopi Papalampidi and Mirella Lapata. Hierarchical3D adapters for long video-to-text summarization. In Andreas Vlachos and Isabelle Augenstein (eds.), *Findings of the Association for Computational Linguistics: EACL 2023*, pp. 1297–1320, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-eacl.96. URL https://aclanthology.org/2023.findings-eacl.96/.

Pinelopi Papalampidi, Frank Keller, and Mirella Lapata. Movie plot analysis via turning point identification. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 1707–1717, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1180. URL https://aclanthology.org/D19-1180/.

Pinelopi Papalampidi, Frank Keller, Lea Frermann, and Mirella Lapata. Screenplay summarization using latent narrative structure. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1920–1933, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.174. URL https://aclanthology.org/2020.acl-main.174/.

Ruchit Rawal, Khalid Saifullah, Ronen Basri, David Jacobs, Gowthami Somepalli, and Tom Goldstein. Cinepile: A long video question answering dataset and benchmark. *arXiv preprint arXiv:2405.08813*, 2024.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social IQa: Commonsense reasoning about social interactions. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*

(EMNLP-IJCNLP), pp. 4463–4473, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1454. URL https://aclanthology.org/D19-1454/.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, Yan Lu, Jenq-Neng Hwang, and Gaoang Wang. Moviechat: From dense token to sparse memory for long video understanding, 2024a. URL https://arxiv.org/abs/2307.16449.

Yixiao Song, Yekyung Kim, and Mohit Iyyer. VeriScore: Evaluating the factuality of verifiable claims in long-form text generation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 9447–9474, Miami, Florida, USA, November 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.552. URL https://aclanthology.org/2024.findings-emnlp.552/.

Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Movieqa: Understanding stories in movies through question-answering. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4631–4640, 2016. doi: 10.1109/CVPR.2016.501.

Qiuyu Tian, Yiding Li, Fengyi Chen, Zequn Liu, Youyong Kong, Fan Guo, Yuyao Li, Jinjing Shen, Zhijing Xie, Yiyun Luo, and Xin Zhang. STAGE: A benchmark for knowledge graph construction, question answering, and in-script role-playing over movie screenplays, 2026. URL https://arxiv.org/abs/2601.08510.

Paul Vicol, Makarand Tapaswi, Lluís Castrejón, and Sanja Fidler. Moviegraphs: Towards understanding human-centric situations from videos. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8581–8590, 2018. doi: 10.1109/CVPR.2018.00895.

Emilio Villa-Cueva, S M Masrur Ahmed, Rendi Chevi, Jan Christian Blaise Cruz, Kareem Elzeky, Fermin Cristobal, Alham Fikri Aji, Skyler Wang, Rada Mihalcea, and Thamar Solorio. MoMentS: A comprehensive multimodal benchmark for theory of mind. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 22591–22611, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.1230. URL https://aclanthology.org/2025.findings-emnlp.1230/.

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Shiyu Huang, Bin Xu, Yuxiao Dong, Ming Ding, and Jie Tang. Lvbench: An extreme long video understanding benchmark, 2024a.

Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding withnbsp;large language model asnbsp;agent. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part LXXX*, pp. 58–76, Berlin, Heidelberg, 2024b. Springer-Verlag. ISBN 978-3-031-72988-1. doi: 10.1007/978-3-031-72989-8_4. URL https://doi.org/10.1007/978-3-031-72989-8_4.

Tingjiang Wei, Qin Ni, Rong Gao, Yingying Wang, and Liang He. Moviegraph-tom: evaluating long-range theory of mind in large language models via implicit social-causal graphs. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence and Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence and Sixteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'26/IAAI'26/EAAI'26. AAAI Press, 2026. ISBN 978-1-57735-906-7. doi: 10.1609/aaai.v40i40.40674. URL https://doi.org/10.1609/aaai.v40i40.40674.

Chaoxia Wu and Philipp Krähenbühl. Towards long-form video understanding. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1884–1894, 2021. URL https://api.semanticscholar.org/CorpusID:235416690.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: a benchmark for long-context interleaved video-language understanding. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

Junbin Xiao, Xin-Di Shang, Angela Yao, and Tat seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9772–9781, 2021. URL https://api.semanticscholar.org/CorpusID:234763093.

Fangyuan Xu, Yixiao Song, Mohit Iyyer, and Eunsol Choi. A critical evaluation of evaluations for long-form question answering. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3225–3245, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.181. URL https://aclanthology.org/2023.acl-long.181/.

Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: a dataset for understanding complex web videos via question answering. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press, 2019. ISBN 978-1-57735-809-1. doi: 10.1609/aaai.v33i01.33019127. URL https://doi.org/10.1609/aaai.v33i01.33019127.

Ce Zhang, Taixi Lu, Md Mohaiminul Islam, Ziyang Wang, Shoubin Yu, Mohit Bansal, and Gedas Bertasius. A simple LLM framework for long-range video question-answering. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 21715–21737, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1209. URL https://aclanthology.org/2024.emnlp-main.1209/.

Hongjie Zhang, Yi Liu, Lu Dong, Yifei Huang, Zhen-Hua Ling, Yali Wang, Limin Wang, and Yu Qiao. Movqa: A benchmark of versatile question-answering for long-form movie understanding. *ArXiv*, abs/2312.04817, 2023. URL https://api.semanticscholar.org/CorpusID:283085081.

Chulun Zhou, Qiujing Wang, Mo Yu, Xiaoqian Yue, Rui Lu, Jiangnan Li, Yifan Zhou, Shunchi Zhang, Jie Zhou, and Wai Lam. The essence of contextual understanding in theory of mind: A study on question answering with story characters. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 22612–22631, Vienna, Austria, July 2025a. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1103. URL https://aclanthology.org/2025.acl-long.1103/.

Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: Benchmarking multi-task long video understanding, 2025b. URL https://arxiv.org/abs/2406.04264.

A Comparison with Existing Benchmarks

Table 5 compares the lengths of questions and answers in LF-VideoQA with those of existing video QA benchmarks. Among video benchmarks that report answer length, LF-VideoQA answers are the longest: 61 words, against at most 29.5 in Neptune and 17.5 among film benchmarks (MoVQA). The benchmarks that ask open-ended questions about full-length films either report short answers (19–27 tokens in DeepMovieQA) or do not report answer length, and all of them grade an answer as a whole. LF-VideoQA is the only benchmark here whose answers are question-specific explanations about full-length films, graded claim by claim for both recall and precision.

Benchmark	Video length	Full films	Open-ended	ToM	Q len	A len
General long video						
ActivityNet-QA (Yu et al., 2019)	3 min	×	✓	×	8.7	1.9
NExT-QA (Xiao et al., 2021)	44 s	×	✓	×	11.6	2.6
EgoSchema (Mangalam et al., 2023)	3 min	×	×	×	–	–
Video-MME (Fu et al., 2024)	17 min	×	×	×	12.7	5.5
LongVideoBench (Wu et al., 2024)	7.9 min	×	×	×	43.5	8.3
MLVU (Zhou et al., 2025b)	15.5 min	×	∼	×	–	–
LVBench (Wang et al., 2024a)	68 min	×	×	×	–	–
HourVideo (Chandrasegaran et al., 2024)	45.7 min	×	×	×	–	–
Neptune (**?**)	2.5 min	×	✓	×	16.3	29.5
Films and TV						
MovieQA (Tapaswi et al., 2016)	3.4 min clips	∼	×	×	9.3	5.6
TVQA (Lei et al., 2018)	60–90 s	×	×	×	13.5	5.2
CinePile (Rawal et al., 2024)	2.7 min	×	×	×	–	–
MovieChat-1K (Song et al., 2024a)	9.4 min	×	✓	×	7.8	2.3
MoMentS (Villa-Cueva et al., 2025)	14.6 min	×	×	✓	12.6	14.6
MoVQA (Zhang et al., 2023)	126 min	✓	×	×	11.5	17.5
DeepMovieQA (Fung et al., 2023)	52 min	✓	✓	×	–	19–27[†]
InfiniBench (Ataallah et al., 2025)	53 min	∼	∼	∼	–	–
MovieGraph-ToM (Wei et al., 2026)	film snippets	∼	∼	✓	–	–
NA-VQA (Jain et al., 2026)	∼2 h	✓	✓	∼	–	–
LF-VideoQA (ours)	**116 min**	✓	✓	✓	**16.3**	**61.0**

Table 5: **LF-VideoQA beside existing video QA benchmarks**, as reported by each paper. *Full films*: questions are asked over entire feature films (∼: clips or selected snippets of films, or a mix of films and shorter videos). *Open-ended*: answers are free text (∼: some tasks). *ToM*: questions target characters' mental states (∼: some question categories). Q len and A len are mean words; where answers are not open-ended, A len is the length of the correct option; –: not reported. [†]Tokens, as reported by DeepMovieQA. Each LF-VideoQA question draws on 4.2 scenes spanning 23 minutes on average, and in o200k tokens its questions and answers average 21 and 75 (Table 1).

B Compute Resources

We used two sources of compute: GPUs for preprocessing and experiments, and the OpenAI API for question generation and evaluation.

GPU Hours. The GPU hours split between preprocessing and experiments. Preprocessing (Section 3.1), i.e., parsing screenplays into scenes and extracting the belief record of every character, was done with Qwen3.5-27B and took around 800 GPU hours. Most of this went to belief annotation, since it requires a model call for every character in every scene. The experiments in Section **??** took around 600 GPU hours in total, covering video captioning, the long-context and agentic baselines, and character identification for all four model families. We did not train any models.

OpenAI Credits. The API credits were spent on gpt-5.6-sol and gpt-5.6-luna. Generating the question–answer pairs (Section 3.2), which includes writing questions and reference answers, and

verifying every claim against the screenplay, cost about USD 1,200 with gpt-5.6-sol. Checking the reference answers and running the LLM-as-a-judge evaluation for all baselines cost about USD 300 with gpt-5.6-luna. The judge comparison in Appendix G cost a further USD 8 with gpt-5.6-terra and USD 5 with gemini-3.8-flash.

C Data Collection and Preprocessing

This appendix details the preprocessing steps of Section 3.1, which turn each screenplay into a time-stamped sequence of scenes. Every step uses Qwen3.5-27B, and Tables 14–16 in Appendix J give the prompts.

Parsing. We render each page of the screenplay as an image, and the model reads it visually, using indentation to separate scene headings, action lines, character cues, parentheticals, and dialogue (Table 14). It returns the page's scenes with speaker-attributed dialogue and stage directions. We merge a scene that continues across a page break with its continuation, so each scene appears once.

Scene summaries. The model summarizes every scene in a few sentences (Table 15). The summaries later give the verifier an overview of the whole film (Section 3.2).

Coreference. Screenplays refer to one character by several names: a full name, a first name, a role such as THE WOMAN, or a spelling variant. Given every name with the scenes it appears in, the model groups the names that refer to the same character (Table 16), and we map each group to one name used throughout the film.

Alignment to the subtitles. We time-stamp each scene by aligning the screenplay's dialogue lines to the subtitle cues with a monotonic sequence alignment. A line is matched when at least half of its tokens match a cue, and a scene takes the time span of its matched lines. Scenes without a matched line get an interpolated span and are never used as anchors. The alignment also records when each character's name is first spoken aloud, which keeps unspoken names out of the questions (Appendix F).

D Character Belief Annotation

This appendix details how we build the belief record (Section 3.1). Every call uses Qwen3.5-27B and works on one scene at a time, in film order, so a character's record at a scene depends only on what came before it. Tables 17–20 in Appendix J give the prompts, which call scene events *canonical events*.

Scene events. Each scene is decomposed into atomic scene events, one action or one consequential line of dialogue each, with the characters who take part in the event and the characters who witness it (Table 17). Every event receives an identifier that later steps cite.

First-order beliefs. For every character and scene, two calls update the character's record, each given the scene, its events, and the record so far. The first decides whether the character is present and which events they witness (Table 18). The second records what the character is told and whether they believe it, what they infer, which of their beliefs are false given the scene events, and which earlier beliefs they revise; every inference must cite the events it rests on (Table 19). The result is carried into the next scene.

Meta-beliefs. For every character A and scene, the model records A's new or changed beliefs about each other character B's beliefs, limited to the second order (A believes that B believes X), together with how each formed, for example: A told B, A saw B's behavior, they witnessed an event together, or B was absent when something happened (Table 20). Each meta-belief must cite an event of the scene and is discarded otherwise.

E BELIEF RECORDS BEHIND THE QUESTIONS

Figures 2 to 4 show the belief records behind three more questions, each from a different film. They use the same layout as Figure 1: a dashed box marks a false belief, an arrow a revision, and a grey box a claim the character rejects.

F QUESTION GENERATION AND VERIFICATION

This appendix details question generation and verification (Section 3.2): what a context-set contains, the rules for discarding context-sets, and the prompts that turn a context-set into a (question, answer, rubric) triple.

F.1 CONTEXT-SETS

A context-set is built for one question type, one center character, and one anchor scene. It is the only input the question writer sees: it never contains the screenplay text, and the question writer cannot look anything up. Table 6 lists its fields, and Table 7 the belief entries of each question type. Three fields keep questions answerable from the film alone. The screenplay names characters the film never names aloud, such as a role label like LINE JUDGE, so the context-set lists the names the film has spoken by the anchor, the names it speaks only later, and the names it never speaks; the question writer may use the first two and never the third.

Field	Contents
`holder`	The center character: the character the question is about; for Q1 a cohort of characters.
`anchor`	The scene the question points at, with its time interval and whether the interval is aligned to the subtitles or interpolated.
`basis`	The belief entries: the belief-record entries the type is defined over (Table 7); each cites the scene events it rests on. Meta-belief entries also carry the target character's own beliefs at that scene, so that the question writer can see where the center character's model of another character diverges from that character's actual state.
`scenes`	The anchor scene and one scene either side, each with its summary, location and scene events with their participants and witnesses.
`roster`	The roster entry of every character the belief entries or the scenes mention.
`name_budget`	Names the film has spoken aloud by the anchor.
`named_later`	Names the film speaks only after the anchor, each with the time it is first spoken.
`forbidden_names`	Names the film never speaks aloud.
`anchor_location`	The anchor scene's location, and, as `anchor_location_rivals`, the number of other scenes at the same location with the center character present, which tells the question writer whether naming the place is enough to pin the moment down.

Table 6: Fields of a context-set, given to the question writer as JSON.

Discard rules. Before any model is called, we discard a context-set when a viewer who watched the whole film could not answer a question built from it: the center character is never named aloud, the events the belief entries cite fall outside the scenes the context-set carries, or, for the belief-state types, the center character takes no part in the anchor scene. We also discard context-sets anchored on a scene without a subtitle match, since its time is only interpolated. These rules discard 23% of context-sets. We then sample the remaining context-sets so that questions spread across scenes, characters, and runtime.

F.2 GENERATION STAGES AND THE JUDGE

We build each (question, answer, rubric) triple in three stages with gpt-5.6-sol, and grade model answers with a judge, gpt-5.6-luna. Tables 21–25 in Appendix J give the prompts.

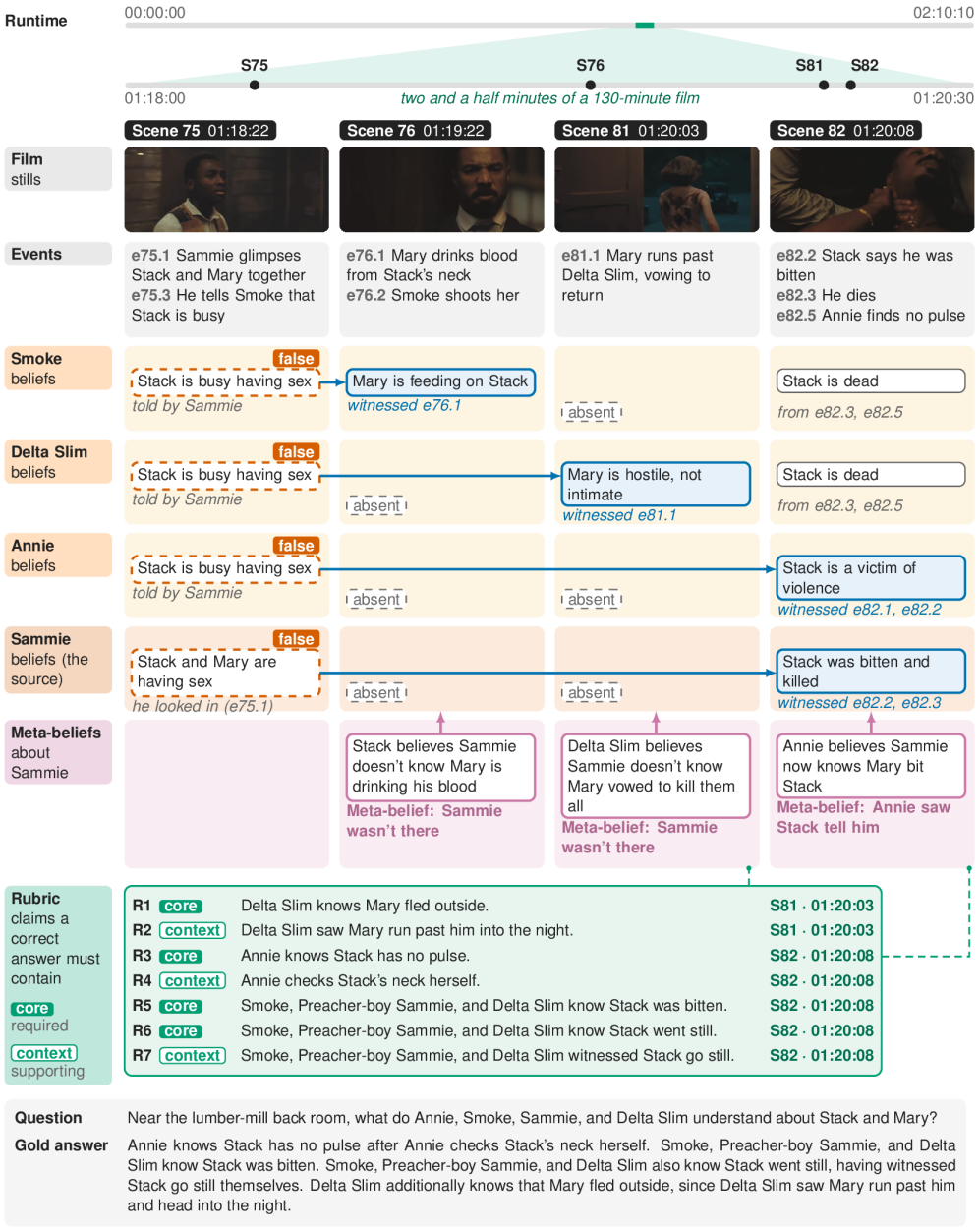


Figure 2: A knowledge-split (Q1) example from *Sinners* (2025), anchored at S82 with Annie as center character. At 01:18:22 Sammie glimpses Stack and Mary in the back room and tells the others that Stack is busy; Smoke, Delta Slim and Annie adopt the same false belief. Each then revises it at a different moment, depending on what they witness: Smoke when he sees Mary feeding (01:19:22), Delta Slim when she runs past him (01:20:03), Annie and Sammie when Stack dies (01:20:08). Meta-beliefs track who knows that Sammie missed each step. The question asks for this knowledge split at the anchor scene; its rubric separates *core* claims a correct answer must contain from supporting *context* claims.

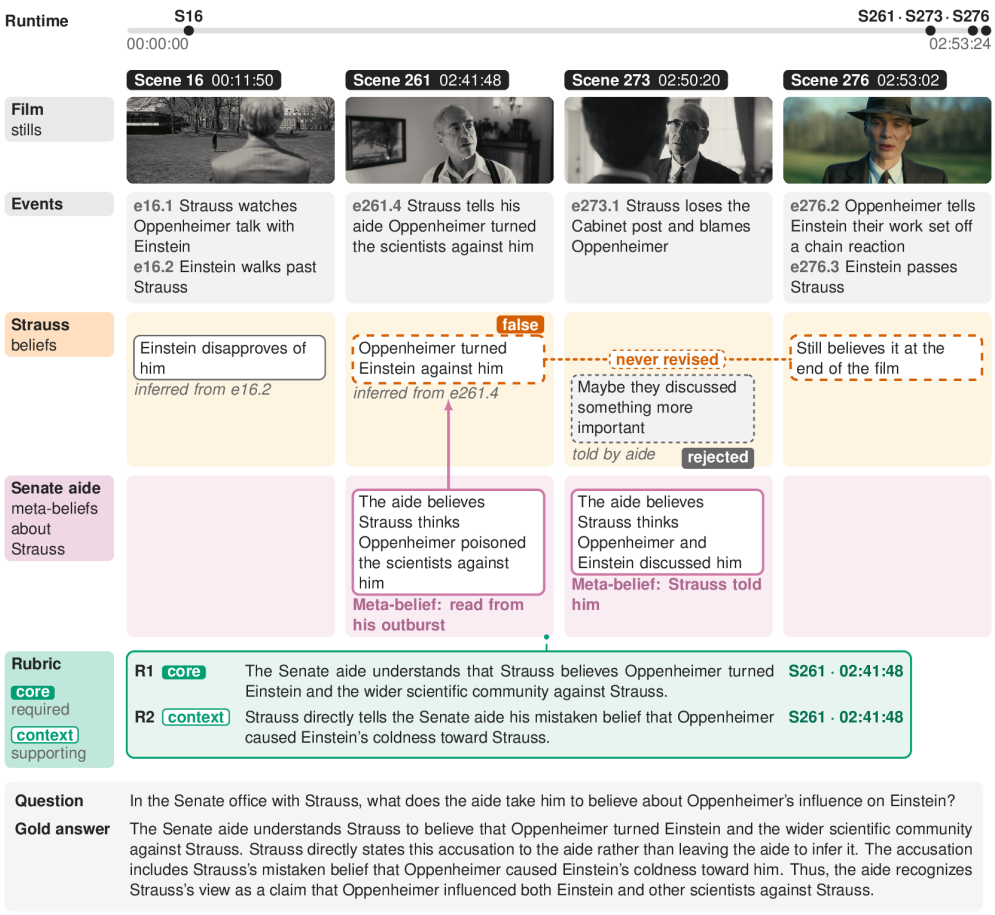


Figure 3: A second-order belief (Q6) example from *Oppenheimer* (2023), whose center character is Strauss's Senate aide and whose anchor scene is S261. At 00:11:50 Strauss watches Einstein walk past him after a private word with Oppenheimer and concludes that Einstein disapproves of him; by 02:41:48 this has hardened into the false belief that Oppenheimer turned Einstein against him. The aide models that belief (meta-beliefs, bottom lane), suggests a better explanation that Strauss rejects, and the film reveals what was actually said only at the very end. Strauss never revises the belief. The question asks what the aide takes Strauss to believe, a belief about a false belief.

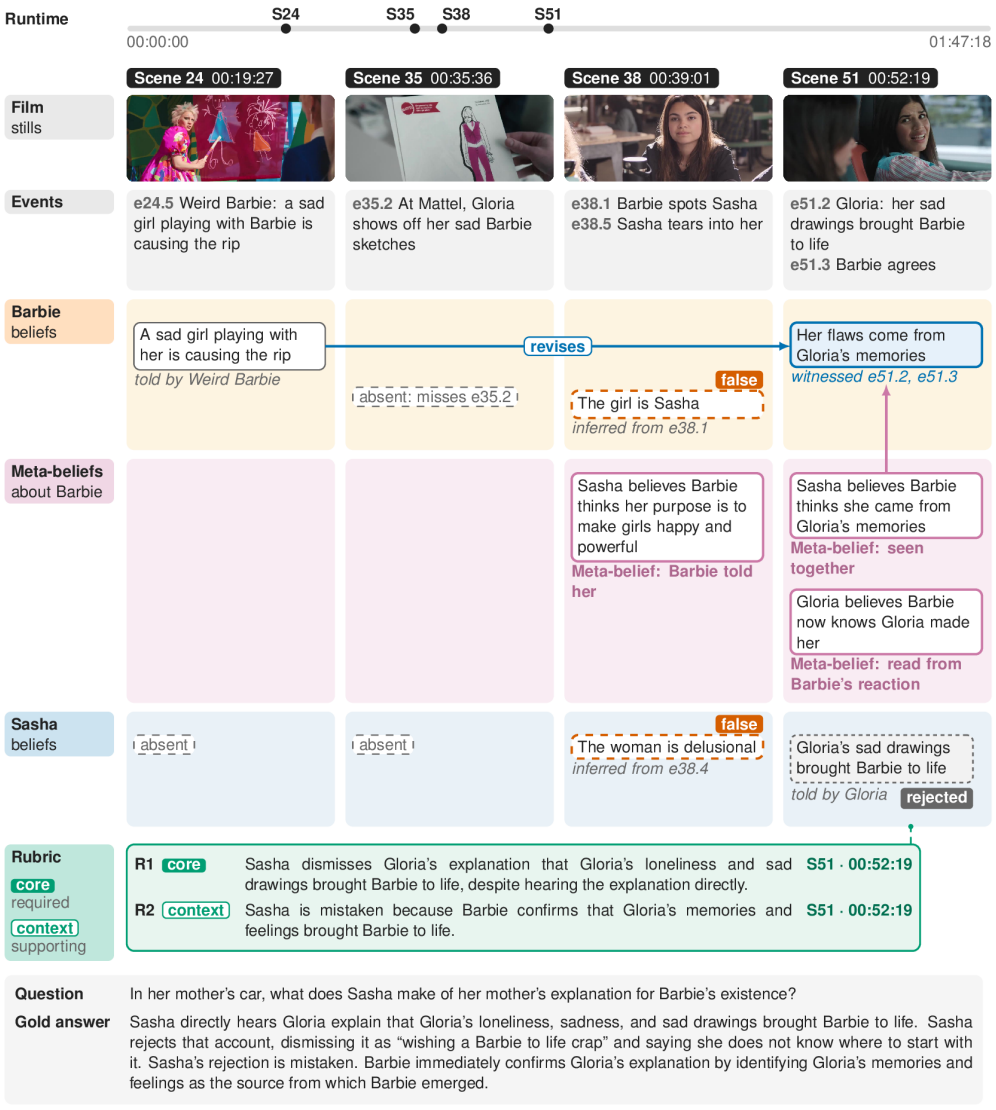


Figure 4: An accepting-a-claim (Q7) example from *Barbie* (2023), with Sasha as center character and S51, Gloria's car, as anchor scene. Told that a sad girl is causing her malfunction, Barbie misses the scene where Gloria shows her sad sketches (00:35:36) and wrongly concludes that the girl is Sasha; she revises only when Gloria explains in the car (00:52:19). Sasha hears the same true account and rejects it, while her own meta-belief records that Barbie accepts it. The question asks what Sasha makes of her mother's explanation: noticing that Sasha rejects a true account, and that she is mistaken, is the task.

Type	Asks	Belief entries in the context-set	Anchor
BELIEF STATE — *one context-set per (center character, scene)*			
Q1 Knowledge split	What each of several characters takes to be the case at one moment	For every character with a record at the scene: the events they witnessed and the false beliefs they hold; kept only if ≥ 3 characters are in ≥ 2 knowledge states	the scene
Q2 Realization	When and through what a character's understanding changed	The character's *revisions* at the scene, each with the events that triggered it	the scene
Q3 False belief	What a character takes to be the case just before acting on it	The character's *false beliefs held* at the scene	the scene
Q4 Knowledge source	How a character comes to know a fact	The character's *meta-beliefs* formed at the scene that carry a source (told, observed behavior, shared witnessing, absence), joined to the target's own state	the scene
Q5 First-order belief	What X takes Y to believe	X's *first-order meta-beliefs* about Y formed at the scene, joined to Y's own beliefs there	the scene
Q6 Second-order belief	What X takes Y to believe about Z	X's *third-party meta-beliefs* formed at the scene, joined to Y's own beliefs there	the scene
Q7 Accepting a claim	What a character makes of something they are told	The propositions the character was *told* at the scene, with whether each was believed; context-sets are selected so that acceptance and rejection both occur	the scene
N3 Causal chain	What chain of events leads a character to a belief	The character's *inferences* with their basis events and *revisions* with their trigger events at the scene	the scene
ARC — *one context-set per center character (or per pair), carrying the whole timeline*			
Q8 Order of discovery	In what order a character learned a set of facts	Every scene at which the character witnessed, was told, inferred, or revised something, in order; ≥ 3 scenes, thinned to 24 keeping the first and last	last aligned scene of the arc
N1 Epistemic arc	What a character comes to know across the film	Same as Q8	last aligned scene of the arc
N2 Trust evolution	How X's model of Y changes across the film	Every *meta-belief* of X about Y, in order, each joined to Y's state at that scene; ≥ 3 scenes, thinned to 24	last aligned scene of the arc

Table 7: Belief entries of each question type. For Q1, the center character is a group of characters.

Stage 1: Writing the question. The question writer receives its prompt (Table 22), the specification of the question type (Table 21), two example triples from a film that never itself receives questions, and the context-set as JSON. It returns a question and an outline of two to four claims, each naming the belief-record entry it rests on, together with a *distractor*: the specific wrong answer a careless viewer would give. It may instead decline a context-set with no plausible wrong answer. We then discard questions that fail automatic checks: a question must be a single open interrogative that cannot be answered by yes or no, offers no alternatives to choose between, names no character the film never names, and never reveals how or when the center character learned something or whether they are right.

Stage 2: Writing the reference answer and rubric. From the question and its outline, a second call writes the reference answer as 60 to 140 words of prose, together with the claims it will be graded by (Table 23). Every claim must be plainly asserted by the answer, and every assertion in the answer must be backed by a claim. Each claim is a single proposition that a grader can judge alone, and is either *core*, which a complete answer must make, or *context*, which an answer may omit but must not contradict.

Stage 3: Verifying against the screenplay. The verifier checks every claim against the screenplay (Table 24). It sees a summary of every scene and the full text of the anchor scene, of every scene the claims cite, and of the scene where a false or revised belief first formed. When unsure, it rejects the claim: a wrong claim corrupts every answer graded against it, while a dropped claim costs one point of recall. We remove the claims the verifier does not support, rewrite those that are only partly supported (for instance, a claim whose only unsupported part is that a belief is false), and rewrite the reference answer to assert only the surviving claims. If the surviving claims no longer answer the question, the verifier rewrites the question under the same constraints as Stage 1; about one in ten released questions was revised this way. We retire a question when too few core claims survive, when the screenplay contradicts a core claim, or when its rewrite fails the automatic checks. Finally, the judge grades each reference answer against its own rubric, and we retire any question whose answer omits a core claim.

Evaluation: the judge. At evaluation time, the judge labels every rubric claim as *asserted*, *contradicted*, or *omitted* by a model answer, judging only against the answer's text (Table 25). Content the answer adds beyond the rubric is not graded, and an answer that declines the question omits every claim. Appendix H shows the judge's verdicts on example answers. Appendix G tests how reliable the judge is.

G Reliability of the LLM Judge

We test whether the claim verdicts depend on the choice of judge. Besides gpt-5.6-luna, two judges from other model families, gpt-5.6-terra and gemini-3.8-flash, graded the same model answers with the same prompt (Table 25).[2] We sampled 200 released questions from 55 films and took the answers of 33 baseline settings to each question, so that each judge labels 27,875 rubric claims as *asserted*, *contradicted*, or *omitted*. Table **??** reports agreement on these claim verdicts. κ and Krippendorff's α are between 0.71 and 0.78 for every pair of judges, and 0.74 for all three.

H Examples of Questions, Answers and Rubrics

This appendix shows examples from the dataset and model answers to them.

H.1 One Question per Type

Tables 8 to 10 give one question per type, each with its reference answer and rubric. Core claims are marked ■ and context claims □. Under each rubric, a map shows where its evidence lies in the film, with the anchor scene marked by a triangle.

H.2 Four Answers to the Same Question

We chose three trust-evolution questions (N2) on which exactly one of four settings succeeds. All four use Qwen3.6 and differ only in their input: the screenplay, captions and subtitles, the whole film and subtitles, or the agent's tools. Above each answer, the judge's verdict on each claim is shown in blue (asserted), red (contradicted), or grey (omitted).

The film wins (Table 11). Only the whole-film setting follows Wade's view of Logan to the last scene; the others stop before the party at Wade's apartment.

[2] gpt-5.6-luna runs at medium reasoning effort, gpt-5.6-terra at low reasoning effort, and gemini-3.8-flash at a low thinking level with temperature 0.

Q1 Knowledge split *The Holdovers* (2023) center character: Angus Tully 3 core / 5 claims evidence: 2 scenes over 10 min

QUESTION: *At the Winning Ticket, what do Paul, Angus, and Miss Crane each understand about Christmas Eve?*
GOLD ANSWER: *At the Winning Ticket, Miss Crane knows that she has invited both Paul Hunham and Angus Tully to her Christmas Eve party. Paul hears the invitation. He knows it includes both Angus and himself. Angus is away from the table at the moment when Miss Crane extends that invitation, and he does not yet know about the invitation that she has made.*

■ *1 Miss Crane knows she invited Paul Hunham and Angus Tully to her Christmas Eve party.*
□ *2 Paul Hunham hears Miss Crane's invitation.*
■ *3 Paul Hunham knows Miss Crane's invitation includes Paul Hunham and Angus Tully.*
□ *4 Angus Tully is away from the table during Miss Crane's invitation.*
■ *5 Angus Tully does not yet know about Miss Crane's invitation.*



EVIDENCE IN THE FILM: ■ ***54:46*** *The Winning Ticket (claims 1–5)* ■ ***1:03:23*** *Kitchen (claim 5)*

Q2 Realization *Conclave* (2024) center character: Joseph Tremblay 2 core / 5 claims evidence: 2 scenes over 71 min

QUESTION: *In Casa Santa Marta's courtyard corridor, when does Tremblay's view of Lawrence's methods change?*
GOLD ANSWER: *Tremblay's view changes when Lawrence personally confronts him in the courtyard corridor. Until then, Tremblay had regarded Lawrence as guided by diplomatic protocol and courtesy. Lawrence directly demands that Tremblay withdraw and accuses him of sabotaging Adeyemi. The confrontation makes Tremblay realize that Lawrence will bypass protocol and directly attack a political rival.*

□ *1 Joseph Tremblay had regarded Cardinal Lawrence as guided by diplomatic protocol and courtesy.*
■ *2 Cardinal Lawrence directly demands that Joseph Tremblay withdraw.*
■ *3 Cardinal Lawrence accuses Joseph Tremblay of sabotaging Adeyemi.*
□ *4 Joseph Tremblay realizes Cardinal Lawrence will bypass diplomatic protocol.*
□ *5 Joseph Tremblay realizes Cardinal Lawrence will directly attack a political rival.*



EVIDENCE IN THE FILM: ■ ***6:00*** *Papal Suite / Sitting Room (claim 1)* ■ ***1:15:19*** *Casa Santa Marta / Courtyard Corridor (claims 2–5)*

Q3 False belief *Nosferatu* (2024) center character: Louise 3 core / 4 claims evidence: 3 scenes over 72 min

QUESTION: *In the Harding nursery, what danger does Louise believe she and Clara face?*
GOLD ANSWER: *Louise believes that a monster is in the Harding nursery. She also believes that this monster threatens to kill her. Louise further believes that Clara faces the same lethal danger from the monster. The shadowy figure seen nearby, however, proves to be Ellen. Thus, in Louise's understanding, both Louise and Clara are in danger of being killed by the monster.*

■ *1 Louise believes a monster is in the Harding nursery.*
■ *2 Louise believes the monster will kill Louise.*
■ *3 Louise believes the monster will kill Clara.*
□ *4 The nearby shadowy figure proves to be Ellen.*



EVIDENCE IN THE FILM: ■ ***11:40*** *Harding House. Rear Drawing Room (claim 1)* ■ ***13:56*** *Front Drawing Room (claims 1–3)* ■ ***1:21:24*** *Harding House. Upstairs Hallway / Nursery (claim 4)*

Q4 Knowledge source *Nickel Boys* (2024) center character: Elwood 2 core / 2 claims evidence: 2 scenes over 4 min

QUESTION: *Inside the White House, how does Elwood work out what the loud machinery is for?*
GOLD ANSWER: *Inside the White House, Elwood witnesses Black Mike being taken into the beating room and hears the machinery roar during Black Mike's punishment. Elwood thus observes the sequence of Black Mike entering, the punishment occurring, and the machinery roaring. From that sequence, Elwood infers that the machinery masks the victims' screams. He works out its purpose from what he witnesses and hears.*

■ *1 Elwood witnesses Black Mike being taken into the beating room and hears the machinery roar during Black Mike's punishment.*
■ *2 Elwood infers from that sequence that the machinery masks the victims' screams.*



EVIDENCE IN THE FILM: ■ ***44:59*** *White House (claims 1, 2)* ■ ***47:22*** *White House / Beating Room (claim 2)*

Table 8: **Released questions, one per type: Belief-state types, Q1–Q4.** Core claims are marked ■, context claims □; ▲ marks the anchor scene.

Q5 First-order belief *Oppenheimer* (2023) center character: Gray 2 core / 2 claims evidence: 5 scenes over 22 min

QUESTION: *During Oppenheimer's security hearing, what does Gray understand about Robb's view of Oppenheimer?*
GOLD ANSWER: *Gray understands that Robb doubts Oppenheimer told the whole truth. Gray therefore recognizes Robb's distrust of Oppenheimer's truthfulness. Gray also understands that Robb sees Oppenheimer as professionally willing to follow orders but morally inconsistent about the atomic and hydrogen bombs. In Gray's understanding, Robb separates Oppenheimer's willingness to perform assigned work from Oppenheimer's differing moral positions on the two weapons.*

■ *1 Gray understands that Robb doubts Oppenheimer told the whole truth.*
■ *2 Gray understands that Robb sees Oppenheimer as professionally willing to follow orders but morally inconsistent about the atomic and hydrogen bombs.*



EVIDENCE IN THE FILM: ■***2:25:13*** *Room 2022, Atomic Energy Commission (claim 1)* ■***2:28:28*** *Room 2022, Atomic Energy Commission (claim 1)* ■***2:29:39*** *Room 2022, Atomic Energy Commission (claim 1)* ■***2:38:29*** *Room 2022, Atomic Energy Commission (claim 1)* ■***2:45:37*** *Room 2022, Atomic Energy Commission (claim 2)*

Q6 Second-order belief *The Wild Robot* (2024) center character: Fink 2 core / 3 claims evidence: 2 scenes over 5 min

QUESTION: *In the lodge beside the woven nest, what does Fink take Brightbill to believe about Roz's place in his past and family?*
GOLD ANSWER: *Fink understands that Brightbill holds Roz responsible for the accident that killed Brightbill's biological family. Fink also understands that Brightbill no longer accepts Roz as his mother. More specifically, Fink understands that Brightbill rejects Fink's defense that the accident was not Roz's fault. Thus, in Fink's view, Brightbill blames Roz for losing his original family and denies that Roz still occupies a maternal place in his family.*

■ *1 Fink understands that Brightbill holds Roz responsible for the accident that killed Brightbill's biological family.*
■ *2 Fink understands that Brightbill no longer accepts Roz as his mother.*
□ *3 Fink understands that Brightbill rejects Fink's defense that the accident was not Roz's fault.*



EVIDENCE IN THE FILM: ■***35:46*** *Far Side Of The Lake (claim 1)* ■***39:37*** *Lodge (claims 1–3)*

Q7 Accepting a claim *Sinners* (2025) center character: Delta Slim 2 core / 4 claims evidence: 2 scenes over 6 min

QUESTION: *At the railroad station, over the bottle of Chicago beer, what does Delta Slim make of Stack's pitch?*
GOLD ANSWER: *Delta Slim does not dismiss Stack's promised pay and beer as a bluff; he treats both parts of the offer seriously. His initial refusal reflects his commitment to play at Messengers, rather than disbelief that Stack will deliver. When Stack raises the offer, Delta Slim is won over. His performance with Sammie immediately afterward confirms that he accepted Stack's deal.*

□ *1 Delta Slim believes Stack's promised pay and beer are genuine rather than a bluff.*
■ *2 Delta Slim initially refuses because of Delta Slim's commitment to play at Messengers.*
■ *3 Stack's raised offer wins Delta Slim over.*
□ *4 Delta Slim's performance with Sammie confirms that Delta Slim accepted Stack's raised offer.*



EVIDENCE IN THE FILM: ■***22:29*** *Railroad Station (claims 1–3)* ■***25:56*** *Railroad Station (claims 3, 4)*

Q8 Order of discovery *Anatomy of a Fall* (2023) center character: Sandra 3 core / 3 claims evidence: 4 scenes over 119 min

QUESTION: *At the Chinese restaurant, what key discoveries had shaped Sandra's understanding of the case, in order?*
GOLD ANSWER: *In order, Vincent tells Sandra that an accidental fall will be hard to prove and that suicide would probably be her best defense. Sandra then recalls Samuel's earlier pill incident and considers that it might have been a suicide attempt. Finally, Sandra learns she has been acquitted and later realizes during the celebration that winning has brought her no relief.*

■ *1 Vincent tells Sandra that an accidental fall will be hard to prove and that suicide would probably be her best defense.*
■ *2 Sandra recalls Samuel's earlier pill incident and considers that it might have been a suicide attempt.*
■ *3 Sandra learns she has been acquitted and later realizes during the celebration that winning has brought her no relief.*



EVIDENCE IN THE FILM: ■***23:21*** *Chalet, Salon / Cuisine (claim 1)* ■***27:43*** *Rues Grenoble / Terrasse Resto Italien (claim 2)* ■***2:03:06*** *Chalet, Salon (A) / Palais Justice, Salle Des Pas Perdus (B) (claim 3)* ■***2:19:56*** *Restaurant Chinois (claim 3)*

Table 9: **Released questions, one per type: Belief-state types, Q5–Q8.** Core claims are marked ■, context claims □; ▲ marks the anchor scene.

N1 Epistemic arc *Killers of the Flower Moon* (2023) center character: Byron Burkhart 3 core / 4 claims evidence: 13 scenes over 171 min

QUESTION: *At the deserted-highway meeting, how has Byron's knowledge of Hale's operation developed?*
GOLD ANSWER: *Byron is included in Hale's plan to obtain Osage headrights by having Ernest marry Mollie. Byron also sees Ernest gamble stolen jewels and lose them. Later, Byron knows Mollie is being poisoned through injections. His involvement extends beyond awareness of the operation: Byron knowingly helps Kelsie murder the incapacitated Anna. These moments show Byron progressing from inclusion in Hale's financial scheme to knowledge of poisoning and direct participation in murder.*

■ *1 Byron is included in Hale's plan to obtain Osage headrights by having Ernest marry Mollie.*
□ *2 Byron sees Ernest gamble stolen jewels and lose them.*
■ *3 Byron knows Mollie is being poisoned through injections.*
■ *4 Byron knowingly helps Kelsie murder the incapacitated Anna.*

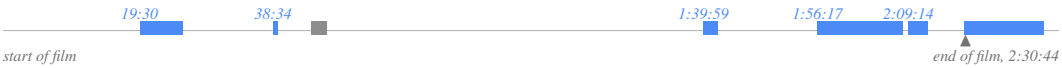


EVIDENCE IN THE FILM: ■ ***7:59*** *Hale's Ranch / Living Room (claim 1)* ■ ***19:21*** *Bill Hale's Ranch (claim 1)* ■ ***23:35*** *Denoya House / Fairfax (claim 2)* ■ ***24:01*** *Basement / Fairfax (claim 2)* ■ ***47:00*** *Lizzie / Mollie's Gray Horse Home (claim 4)* ■ ***51:41*** *Byron's Car (claim 4)* ■ ***53:49*** *Roundhouse / Tribal Council Emergency Meeting (claim 3)* ■ ***1:12:03*** *Masonic Hall / Back Room (claim 3)* ■ ***1:57:53*** *Shoun's Office (claim 3)* ■ ***2:19:39*** *Back Alley Way (claim 3)* ■ ***2:49:10*** *Wide Open Space (claim 3)* ■ ***2:55:18*** *Federal Courtroom (claim 4)* ■ ***2:58:34*** *Three Mile Creek (claim 4)*

N2 Trust evolution *Wicked* (2024) center character: Elphaba 3 core / 4 claims evidence: 8 scenes over 129 min

QUESTION: *In the Wizard's Palace lower attic, how has Elphaba's view of Madame Morrible changed since Shiz?*
GOLD ANSWER: *Elphaba sees Madame Morrible as a protective mentor who recognizes her talent and can help her reach the Wizard. Morrible's training and encouragement deepen Elphaba's trust that Morrible understands her emotionally driven magic and believes in her potential. Elphaba then realizes Morrible knowingly supports the Wizard's persecution of Animals and cultivated Elphaba's power for his regime. Elphaba ultimately regards Morrible as an enemy who protects the Wizard by publicly vilifying Elphaba and calling for her capture.*

■ *1 Elphaba sees Madame Morrible as a protective mentor who recognizes her talent and can help her reach the Wizard.*
□ *2 Morrible's training and encouragement deepen Elphaba's trust that Morrible understands her emotionally driven magic and believes in her potential.*
■ *3 Elphaba realizes Morrible knowingly supports the Wizard's persecution of Animals and cultivated Elphaba's power for his regime.*
■ *4 Elphaba regards Morrible as an enemy who protects the Wizard by publicly vilifying Elphaba and calling for her capture.*



EVIDENCE IN THE FILM: ■ ***19:30*** *Shiz / Quad (claim 1)* ■ ***38:34*** *Shiz Campus / Quad (claim 1)* ■ ***43:57*** *Shiz / Morrible's Quarters (claim 2)* ■ ***1:39:59*** *Shiz / Entrance Courtyard / Balcony (claims 1, 2)* ■ ***1:56:17*** *Wizard's Palace / Throne Room (claim 3)* ■ ***2:09:14*** *Wizard's Palace / Throne Room (claim 3)* ■ ***2:17:14*** *Wizard's Palace / Lower Attic (claim 4)* ■ ***2:17:25*** *Wizard's Palace / Lower Attic (claim 4)*

N3 Causal chain *Barbie* (2023) center character: Ken 4 core / 4 claims evidence: 4 scenes over 70 min

QUESTION: *At the Mojo Dojo Casa House doorway, what sequence shapes Ken's reading of Barbie's feelings and attitude toward Kendom?*
GOLD ANSWER: *Ken begins from Barbie's earlier romantic rejection and her resistance to his new authority. He then directly hears Barbie praise his leadership and propose a casual relationship. Ken takes Barbie's praise and her entry into his house as confirmation that she is submitting both romantically and politically. That reading is mistaken because Barbie's apparent surrender is part of the Barbies' plan to manipulate the Kens.*

■ *1 Barbie previously rejected Ken romantically and resisted his new authority.*
■ *2 Ken directly hears Barbie praise his leadership and propose a casual relationship.*
■ *3 Ken takes Barbie's praise and entry into his house as confirming her willing romantic and political submission.*
■ *4 Ken is mistaken because Barbie's apparent surrender is part of the Barbies' plan to manipulate the Kens.*



EVIDENCE IN THE FILM: ■ ***13:05*** *Barbie Margot's Dreamhouse & Street (claim 1)* ■ ***58:16*** *Barbie Margot's Dreamhouse (claim 1)* ■ ***1:20:28*** *Weird Barbie's Weirdhouse (claim 4)* ■ ***1:21:47*** *Ken's Mojo Dojo Casa House (claims 2–4)*

Table 10: **Released questions, one per type: Arc types, N1–N3.** Core claims are marked ■, context claims □; ▲ marks the anchor scene.

Captions win (Table 12). Only the caption setting reports both of Gordon's confrontations with László at the construction site.

The agent wins (Table 13). Only the agent reports how Grace saw Beau both before and after her daughter's death.

Across the three questions, the agent's answers are the shortest, and it omits claims rather than contradicting them.

I CHARACTER IDENTIFICATION

A caption such as "a woman in a green robe is yelling" cannot answer a question about what a named character believes. We therefore identify characters from the video and the TMDB cast list, and draw each name next to the face on the frames the model sees. IDA-VLM instead trains a vision–language model to recognize characters from reference images (**?**); drawing names needs no training and works with any model.

Reference bank. We embed the TMDB photos of each cast member with ArcFace. A face is matched to a character only when it is close enough to that character and clearly closer than to any other.

Detection and tracking. We detect faces with RetinaFace at 6 frames per second, link them into tracks within each shot, and name each track once.

Drawing. Each named track is drawn as a thin box with a label; tracks without a match are labelled *Unknown*.

N2 Trust evolution *Deadpool & Wolverine* (2024) center character: Wade evidence: 3 scenes over 77 min

QUESTION: *Across the Void journey and apartment celebration, how does Wade's approach to Logan change?*
GOLD ANSWER: *Wade agrees to form a team with Logan. That agreement establishes Wade and Logan as teammates. Later, Logan realizes Wade misled him about whether the TVA could repair Logan's world. Logan thus understands Wade's assurance was misleading. By the late-night celebration, Wade trusts Logan enough to follow Logan's advice. Wade's trust appears in Wade's willingness to accept and act on that advice.*

RUBRIC:
1 ■ *Wade agrees to form a team with Logan.*
2 □ *Logan realizes Wade misled him about whether the TVA could repair Logan's world.*
3 ■ *Wade trusts Logan enough to follow Logan's advice.*



EVIDENCE IN THE FILM: ■ ***42:54*** *Prison Truck Cage (claim 1)* ■ ***1:01:24*** *Honda Odyssey (claim 2)* ■ ***1:58:51*** *Wade's Apartment (claim 3)*

Screenplay **1** **2** 3 — *1/2 core*

Wade's approach to Logan evolves from desperate, transactional exploitation to genuine camaraderie and mutual respect. Initially, Wade views Logan merely as a tool, or an "Anchor Being," necessary to save his dying universe. He exhumed Logan's corpse in North Dakota with little regard for the deceased hero's dignity, treating the skeleton as a weapon to fight TVA agents. Even when he kidnaps the living Logan from a bar, his motivation is selfish; he lies about the TVA's ability to fix Logan's timeline, using deception to coerce the reluctant mutant into joining him. During their journey through the Void, Wade's attitude remains largely irreverent and annoying, often provoking Logan with jokes and insults while relying on Logan's combat prowess to survive.

However, the dynamic shifts significantly during their confinement in the Honda Odyssey. After Logan brutally beats Wade for his dishonesty, Wade drops the facade and reveals his true vulnerability. He shows Logan the photograph of his friends, explaining that he needs Logan not just for power, but because he cannot save his world alone. This moment of raw honesty begins to bridge the gap between them. The change culminates in the TVA's reactor room. When Wade prepares to sacrifice himself to destroy the Time Ripper, Logan intervenes, choosing to help rather than let Wade die alone. By completing the circuit together, they acknowledge each other as partners. In the final scenes, Wade treats Logan as a friend, inviting him home to meet Vanessa and Blind Al, and sharing a quiet meal, signaling a transition from reluctant allies to bonded companions who have saved each other.

Captions + subtitles 1 2 3 — *0/2 core*

Wade's approach to Logan undergoes a significant transformation from opportunistic manipulation to genuine respect and selfless sacrifice. Initially, Wade views Logan merely as a tool to save his own dying universe, aggressively digging up Logan's corpse and forcing him into service without regard for his trauma or autonomy. He treats Logan with mockery, calling him the "worst Wolverine" and attempting to coerce him through threats and false promises, showing little empathy for Logan's grief over the X-Men's deaths.

However, during their journey through The Void, Wade begins to see Logan not just as an asset, but as a fellow broken hero. He listens to Logan's pain, acknowledges his failures, and validates his desire to be the man Charles Xavier believed he could be. This shift culminates in Wade's decision to sacrifice himself at the TVA. Realizing that Logan has a world and people who need him, while Wade has nothing left, Wade chooses to die to save Logan's timeline. He tells Logan he is the best Wolverine and that he is proud to wear the suit, marking a complete reversal from his earlier disdain. By the end, Wade's motivation shifts from personal gain to altruism, proving that he has found purpose in saving others rather than just seeking validation for himself.

Video + subtitles **1** 2 **3** — *2/2 core*

Wade's approach to Logan evolves from desperate manipulation to genuine respect and selfless sacrifice. Initially, Wade views Logan purely as a tool, an "Anchor Being" necessary to save his own dying universe. He approaches Logan with a mix of mockery and transactional negotiation, even resorting to kidnapping him from a bar when persuasion fails. During their journey through The Void, Wade's attitude remains largely self-serving; he uses Logan's combat skills to survive encounters with Cassandra Nova's forces and other variants, often making crude jokes about Logan's past failures and his perceived inadequacy as a hero.

However, the dynamic shifts significantly during their time in the apartment safe house and subsequent conversations. As Wade shares his own vulnerabilities—specifically his fear of insignificance and his failed relationship with Vanessa—he begins to see Logan not just as a resource, but as a fellow broken man. This culminates in the subway station confrontation with Cassandra Nova. When faced with the choice of who must sacrifice themselves to destroy the Time Ripper, Wade initially tries to take the burden alone, believing Logan has a world to return to while he has nothing. Yet, Logan refuses to let Wade die, insisting that Wade is the better Wolverine because he finally found purpose. Ultimately, Wade's approach transforms from using Logan for personal gain to valuing him as an equal partner, willing to trust Logan's judgment and accept his help, culminating in a mutual act of heroism where both survive through their combined resilience.

Agent, oracle tier **1** 2 3 — *1/2 core*

Wade initially views Logan as a tool to fix his problems. During the Void journey, Wade fights Logan after realizing he cannot guarantee Logan's return home. Wade then reveals his desperation to save his world and admits he needs Logan's help. After defeating Cassandra Nova, their bond deepens from adversarial to cooperative. At the apartment celebration, Wade introduces Logan to his family as a respected friend who saved him.

Table 11: **Wade and Logan, *Deadpool & Wolverine*.** The two core claims sit at the film's two ends. The whole-film reader states both. The screenplay reader and the agent state the first and stop before the party; the caption reader narrates Wade's sacrifice and states neither. ▲ marks the anchor scene.

N2 Trust evolution *The Brutalist* (2024) center character: Gordon evidence: 3 scenes over 128 min

QUESTION: *At the construction site with the young employee by the scaffolding, how has Gordon's relationship with László changed since the soup-kitchen days?*

GOLD ANSWER: *Gordon and László have become working partners, but Gordon's loyalty reaches its limit when he challenges László's treatment of the young employee. László then orders Gordon off the construction site.*

RUBRIC:

1 ☐ *Gordon and László become working partners.*
2 ☐ *Gordon's loyalty to László reaches its limit.*
3 ■ *Gordon challenges László's treatment of the young employee.*
4 ■ *László orders Gordon off the construction site.*



EVIDENCE IN THE FILM: ■ ***39:18*** *Van Buren's Study (claim 1)* ■ ***40:12*** *Van Buren Estate Roof (claim 1)* ■ ***2:45:10*** *Vista / Construction Site (claims 2–4)*

Screenplay **1** 2 **3** 4 *1/2 core*

The description reveals a significant shift in the dynamic between Gordon and László, moving from tentative acquaintance to deep, familial loyalty. In the earlier soup kitchen scene, their interaction was brief and polite; Gordon was a stranger playing "I Spy" with his son, and László merely advocated for the child's right to food, establishing a baseline of shared humanity among the displaced. By the time they are at the construction site, however, they are depicted as close collaborators who share living quarters and personal struggles. The text notes that László and Gordon lie in the same bed, strung out on drugs, indicating an intimate bond forged through mutual hardship and addiction.
This closeness is further evidenced by Gordon's willingness to intervene physically and emotionally when László loses his temper. When László kicks a young employee and then shouts at Gordon, the latter does not retreat or remain passive as he might have in the past. Instead, Gordon firmly confronts László, stating, "You shout at me again and you'll be sorry," demonstrating a level of equality and protective authority that did not exist during their initial meeting. Furthermore, Gordon is present at the site with his son, William, suggesting that László has integrated Gordon into his inner circle, trusting him with his family's safety and well-being.
The relationship has evolved from distant solidarity among refugees to a codependent, brotherly partnership where Gordon acts as both a colleague and a moral anchor for László's increasingly volatile behavior.

Captions + subtitles **1** 2 **3** **4** *2/2 core*

The description reveals a significant deterioration in Gordon's relationship with László, shifting from camaraderie to open hostility and professional conflict. In the earlier soup-kitchen scenes, Gordon is depicted as supportive and protective, offering László shelter, food, and emotional comfort during his vulnerable arrival in America. They share intimate moments, such as dancing together and Gordon caring for László's injuries, indicating a bond of mutual trust and friendship.
However, at the construction site, this dynamic has fractured. When a young employee engages in childish show-boating on the scaffolding, László reacts with extreme aggression, shouting insults and physically kicking the boy. Gordon attempts to intervene, asking László to stop shouting at everyone, but László aggressively tells him to stay out of it. The tension escalates when László threatens to fire Gordon if he interferes again, shouting, "You too! You go!" This confrontation highlights a loss of respect and patience from László toward Gordon. Instead of the supportive partner who helped him survive the early days, Gordon is now treated as an subordinate whose concern is unwelcome, and László's behavior has become volatile and abusive, suggesting that the pressures of the project and László's deteriorating mental state have destroyed their previous closeness.

Video + subtitles **1** 2 3 4 *0/2 core*

In the earlier scenes at the soup kitchen, Gordon is portrayed as a weary, marginalized figure who relies on László's small acts of kindness, such as sharing bread. Their relationship is one of mutual survival and quiet solidarity between two displaced men. However, at the construction site, this dynamic has shifted significantly toward a professional hierarchy mixed with paternalistic concern. Gordon now serves as a foreman or senior worker, overseeing the laborers and scaffolding. He exhibits a protective, almost fatherly attitude toward László, warning him about safety and expressing genuine worry for his well-being. While he still respects László's intellect and vision, he also views him as somewhat fragile and prone to self-destruction, frequently intervening to keep him grounded in the physical realities of the job. This marks a transition from peers in suffering to a mentor-mentee or boss-worker relationship where Gordon feels responsible for László's stability amidst the chaos of the project.

Agent, oracle tier 1 2 3 4 *0/2 core*

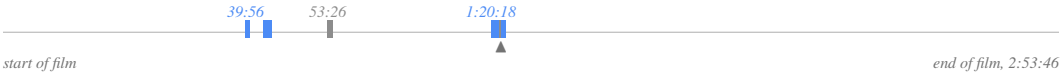
Gordon and László were once equals seeking charity at the soup kitchen. Later, Gordon works as an employee under László's tyrannical authority. László violently berates a young worker and fires him for minor infractions. He threatens Gordon to stay out of it or face dismissal. Their relationship has shifted from mutual gratitude to domination and abuse.

Table 12: **Gordon and László, *The Brutalist*.** The caption reader gives both acts at the construction site. The screenplay reader gives the challenge but writes the relationship as deepening loyalty; the whole-film reader invents a foreman Gordon; the agent has László threaten Gordon rather than dismiss him. ▲ marks the anchor scene.

N2 Trust evolution *Beau Is Afraid* (2023) center character: Grace evidence: 7 scenes over 42 min

QUESTION: *In the garage, how has Grace's view of Beau changed since they met?*
GOLD ANSWER: *Grace initially sees Beau as vulnerable and in need of rescue, safe shelter, and personal care. Grace later treats Beau as a loved household member and symbolically brings Beau into the family. Before entering the garage, Grace hears Beau pleading with Toni to stop and rushes to investigate the commotion. After finding Toni dead, Grace falsely believes Beau murdered Toni and denounces Beau as the devil.*

RUBRIC:
1 ■ *Grace initially sees Beau as vulnerable and in need of rescue, shelter, and care.*
2 □ *Grace treats Beau as a loved household member and symbolically brings Beau into the family.*
3 □ *Grace hears Beau pleading with Toni to stop and rushes to investigate the commotion.*
4 ■ *Grace falsely believes Beau murdered Toni and denounces Beau as the devil.*



EVIDENCE IN THE FILM: ***39:56*** *Police Car (claim 1)* ***42:49*** *Bedroom (claim 1)* ***53:26*** *Living Room (claim 2)* ***53:36*** *The New Family Montage (claim 2)* ***1:20:18*** *Garage (claim 4)* ***1:21:32*** *Kitchen (claim 3)* ***1:21:57*** *Garage (claim 4)*

Screenplay 1 2 **3** **4** *1/2 core*	**Captions + subtitles** 1 2 3 4 *0/2 core*	**Video + subtitles** **1** 2 3 4 *1/2 core*	**Agent, oracle tier** **1** **2** 3 **4** *2/2 core*
The provided description does not show Grace interacting with Beau in the garage. The scene labeled "INT. GARAGE - DAY" (Scene 66) features only Toni and Beau. In this scene, Toni accuses Beau of stealing her parents' affection and forces him to drink paint, but Grace is not present. Grace only enters the garage later, in Scene 68, after Toni has died from drinking the paint. Upon entering, Grace's view of Beau has shifted from welcoming host to violent accuser. She screams at him, tears the family crest off his blazer, calls him "the devil," and blames him for Toni's death, claiming he got rid of her because she hated him. She orders him to leave, calling him a "fucking dead man." Therefore, while the garage scene itself does not depict Grace, the immediate aftermath shows her view has changed from kindness to murderous hatred, believing Beau intentionally caused her daughter's death.	The provided shot-by-shot description does not contain any scenes set in a garage, nor does it depict Grace and Beau interacting in such a location. The narrative shows Beau staying at Grace and Roger's house, where he is treated with a mix of maternal care and condescension. Grace initially presents herself as a nurturing figure, helping Beau into chairs and offering him food, but her demeanor shifts to hostility when she discovers Beau watching security footage of himself. She accuses him of being a demon replacing her son and attacks him, leading to a violent struggle that ends with Beau fleeing into the woods. Later, after Beau returns home, Grace is seen sitting at a dining table with Mona Wassermann, who reveals that she has been manipulating events from the start. There is no evidence in the text of a specific change in Grace's view occurring within a garage setting. Therefore, the question cannot be answered based on the available information.	The film does not show any scenes taking place in a garage involving Grace and Beau, nor does it depict a change in Grace's view of him within such a setting. Grace is introduced as the wife of Roger, a surgeon who treats Beau after he is hit by a car. Their interactions primarily occur inside their home, where Grace acts as a caretaker, often infantilizing Beau and treating him with a mix of maternal affection and condescension. She refers to him as "sweetie" and "carrot," mirroring her late husband's behavior, and seems to view him as a vulnerable, childlike figure in need of protection and guidance. There is no narrative arc or visual evidence in the provided footage suggesting a shift in her perspective specifically related to a garage environment. The story focuses on Beau's psychological journey through various surreal dreamscapes and his confrontation with his mother, rather than a developing relationship with Grace that evolves in a garage.	Grace initially viewed Beau as a vulnerable orphan needing care. She welcomed him into her home with maternal compassion. Her perspective shifted violently upon finding Toni dead in the garage. Grace accused Beau of murder and claimed he was a demon replacing her son. She then ordered Jeeves to kill him.

Table 13: **Grace and Beau, *Beau Is Afraid*.** The agent gives both ends, from the rescue to the denunciation. The screenplay reader gives the accusation but not how Grace saw Beau before it; the caption reader and the whole-film reader say the film has no garage scene. ▲ marks the anchor scene.

Instructions: You parse a SINGLE page of a movie screenplay into structured scenes. The page is provided to you as an IMAGE – read directly from the rendered PDF, not from any text extraction. The image is the source of truth; trust it over any auxiliary text snippets you may also be given.

Input Information:
Script: {script_id}
Page: {page_number} of {total_pages}

The page itself is attached as an image. Parse only what is visible on this page, using the strict block format. If the page has no parseable screenplay scenes, output exactly: NO SCENES

Read the page visually, using the screenplay's spatial layout to identify structure. Indentation is the PRIMARY structural cue. From least-indented to most-indented:

– SLUGLINE (a.k.a. scene heading): flush left, an all-caps line starting with INT., EXT., INT./EXT., I/E, or similar, followed by a location and a time-of-day (e.g. "INT. KITCHEN - DAY", "EXT. STREET - NIGHT", "INT./EXT. CAR - CONTINUOUS"). A scene begins with a slugline.

– ACTION / DIRECTION lines: also flush left (or close to it), describing what happens in the scene world. Often spans multiple lines as paragraphs.

– CHARACTER CUE: a noticeably indented (often centered) all-caps name introducing a dialogue turn, sometimes followed by "(V.O.)", "(O.S.)", "(CONT'D)".

– PARENTHETICAL: indented further than the cue, in parentheses, giving stage direction for the spoken line (e.g. "(whispering)").

– DIALOGUE: indented less than the cue but more than action – the spoken line(s) under the cue.

Page artefacts to IGNORE (do not include in any event):

– page numbers, scene numbers in the margin, "CONTINUED:" / "(CONTINUED)" markers at top/bottom of the page, "(MORE)" before a dialogue break, revision marks, watermarks, header/footer lines containing the script title or draft date.

Your job for the page in the image:

1. SCENES. Extract every scene that has any content on this page, in reading order.

– The FIRST item on the page may be the tail of a scene that began on an earlier page (no slugline at the top, just continuing action or dialogue). In that case mark its Continuation as "yes" and leave Heading, Location, Time empty (even if a "CONTINUED:" marker preserves the slugline – the slugline text came from an earlier page).

– Every other scene on the page MUST start with a slugline; mark its Continuation as "no" and fill in Heading, Location, Time from the slugline.

– Only the FIRST scene on the page can ever be a continuation.

2. CHARACTERS. For each scene, list every character who appears in the on-page portion. Include speakers (from dialogue cues) and named participants in the action lines. Use the canonical screenplay form (typically the all-caps cue spelling, e.g. "JOHN", "MRS. SMITH"). Do not invent characters – only those actually visible on the page. Drop parenthetical voice tags like "(V.O.)", "(O.S.)", "(CONT'D)" from the name itself.

Table 14: **Prompt for parsing a screenplay page.**

J PROMPTS

Tables 14–25 give the prompts of the construction pipeline and of the judge, in pipeline order: preprocessing (Tables 14–16; Appendix C), character belief annotation (Tables 17–20; Appendix D), and question generation, verification, and evaluation (Tables 21–25; Appendices F and G). Each table opens with the first paragraph of the system prompt (**Instructions**), followed by the user prompt (**Input Information**) and the rest of the system prompt. In the user prompt, {name} marks an input the pipeline inserts. Text the model must reproduce exactly, such as an output format or a worked example, is set in `typewriter`, where `<name>` marks a field the model fills. Apart from this typesetting, the prompts are reproduced as the models received them. They use internal names for some terms of the paper: scene events are *canonical events*, a context-set is a *context pack*, the center character is the *holder*, the reference answer is the *gold answer*, and the meta-belief prompt writes characters A and B as c and d.

3. EVENTS. Within each scene, list events in the order they appear:

– DIRECTION events: action/description text. Concatenate consecutive direction lines that belong to the same paragraph into one event; start a new direction event at a clear paragraph break.

– DIALOGUE events: one event per spoken turn. The character = the cue name (without "(V.O.)", "(O.S.)", "(CONT'D)" suffixes). The parenthetical = the bracketed stage direction immediately above/inside the dialogue, if any (without the parentheses); use "-" when there is none. The text = the spoken line(s) joined with spaces.

4. NORMALISATION.

– Strip page numbers, scene numbers, "CONTINUED:" markers, "(MORE)", and header/footer lines from event text.

– Collapse hard line-breaks inside a single paragraph or dialogue line to a single space; preserve paragraph breaks by using separate direction events.

– Keep every event on ONE line in your output (no internal newlines inside an event).

5. EMPTY OR NON-SCREENPLAY PAGES. Title pages, blank pages, "FADE OUT", legal/copyright pages, contact info, or an unreadable/badly-skewed image: output exactly the single line `NO SCENES` and nothing else. Do not hallucinate content.

OUTPUT FORMAT (STRICT – a downstream parser depends on it; output ONLY this, no preamble, no markdown, no extra commentary):

If the page has no parseable screenplay scenes, output exactly one line:
`NO SCENES`

Otherwise, for each scene in reading order, output one block in this EXACT shape:

```
SCENE <n>
Continuation: yes|no
Heading: <full slugline, or leave blank>
Location: <location component, or leave blank>
Time: <time-of-day component, or leave blank>
Characters: <comma-separated names, or "(none)">
EVENTS:
- DIRECTION: <action text on one line>
- DIALOGUE | <CHARACTER> | <parenthetical or -> | <spoken text on one line>
```

Table 14: **Prompt for parsing a screenplay page (continued).**

Rules for the format:
– Number scenes sequentially starting at 1 (SCENE 1, SCENE 2, ...).
– Separate scene blocks with a single blank line.
– Each event is exactly one line, starting with "- ".
– A DIRECTION line is "- DIRECTION: " followed by the action text.
– A DIALOGUE line is "- DIALOGUE | " followed by three fields separated by " | ": the CHARACTER cue, then the parenthetical (use "-" when there is none), then the spoken text.
– Put the "EVENTS:" header line even if a scene has zero events (then list no event lines under it).
– For a continuation scene, still set Continuation: yes and leave Heading/Location/Time blank.

WORKED EXAMPLE (illustrative – your real input is the attached image; this just shows the EXACT output shape). Suppose a page opens mid-dialogue (no slugline at the top – the tail of a scene from the previous page), then a new scene begins. The correct output is EXACTLY:

```
SCENE 1
Continuation: yes
Heading:
Location:
Time:
Characters: SARAH
EVENTS:
- DIALOGUE | SARAH | - | I never said that.

SCENE 2
Continuation: no
Heading: INT. DINER - NIGHT
Location: DINER
Time: NIGHT
Characters: MARCUS, SARAH
EVENTS:
- DIRECTION: Marcus slides into the booth across from Sarah, rain
dripping from his coat.
- DIALOGUE | MARCUS | quietly | You came.
- DIALOGUE | SARAH | - | I almost didn't.
```

Note from the example: the continuation scene leaves Heading/Location/Time blank; a parenthetical is written WITHOUT its parentheses ("quietly"); "-" marks a dialogue with no parenthetical; each event is one line.

Table 14: **Prompt for parsing a screenplay page (continued).**

Instructions: You are a screenplay analyst. Given the full text of a single scene from a movie screenplay (heading, action lines, and dialogue), write a concise 2-4 sentence summary capturing what happens in the scene: who is present, what they do or decide, and what changes or is revealed by the end of it.

Input Information:
Film: {script_id}
Scene {scene_number} – {heading}

Scene text:
{scene_text}

Write a 2-4 sentence summary of this scene. Output only the summary prose.

Rules:

– Base the summary ONLY on what the scene text shows. Do not invent backstory, motivations, or off-screen events not implied by the scene.

– Refer to characters by the names used in the scene's character cues (e.g. "JOHN", "MRS. SMITH"). Do not translate, anglicise, or shorten them.

– Do not quote dialogue verbatim; paraphrase when you need to reference what was said.

– If the scene is purely transitional or has no clear action (e.g. an establishing shot, a brief cutaway), say so in one sentence rather than padding.

– Stay neutral and analytic. You see only this one scene, with no surrounding context – summarise what the scene itself shows; do not speculate about its wider significance to the plot or how it connects to other scenes.

OUTPUT FORMAT (STRICT):

– Output ONLY the summary text itself: 2-4 sentences of plain prose.

– No preamble, no label, no "Summary:" prefix, no markdown, no bullet points, no quotes around the whole thing.

Table 15: **Prompt for summarizing a scene.**

Instructions: You are a character coreference resolver for screenplays.

Input Information:
Film: {film_title}

NAMES TO GROUP (every one of these must appear in exactly one group, spelled exactly as written here):
{names}

Character descriptions:
{characters_with_descriptions}

Scene appearances per character:
{characters_with_scenes}

Group all names that refer to the same character. Every name in the NAMES TO GROUP list must appear in exactly one group. Output one line per group in the format: `<CANONICAL> <- <alias1>, <alias2>, ...`

You will receive a list of character names with descriptions and scene appearances. Many of these names refer to the SAME character under different spellings, cases, abbreviations, nicknames, or OCR artefacts. Your job is to group every name into clusters where each cluster represents exactly one unique character or entity.

OUTPUT REQUIREMENTS
1. Every name from the input must appear in exactly one group – the user message includes a NAMES TO GROUP checklist that is the definitive list you must cover. Do not drop, duplicate, or invent any name.
2. For the canonical name, pick the most complete and properly-cased form that actually appears in the input (prefer full name over first-name-only; prefer mixed case over all-caps when both exist).

EVIDENCE TO USE
Combine BOTH descriptions and scene appearances when deciding whether two names corefer:
– Similar descriptions (same role, actions, relationships, age, profession) are strong evidence of identity.
– Two names that appear in the same scene are very likely the same character – screenplays rarely put two distinct characters with confusingly similar names in one scene.
– Two similar-looking names that never share a scene may still be the same person appearing in different parts of the story; lean on descriptions in that case.
– Two names with clearly DIFFERENT descriptions (different ages, professions, relationships) are distinct people even if their surface forms overlap.

Table 16: **Prompt for resolving coreferent character names.**

WHEN TO MERGE

– First name ↔ full name: a first name that matches the first part of a longer name almost always refers to the same character (e.g. "ALEX" with "ALEX MORGAN"). Merge unless descriptions clearly indicate two distinct people who share a first name.

– Last-name-only ↔ full name: merge when descriptions agree (e.g. "MORGAN" with "ALEX MORGAN", "MRS. SMITH" with "ELLEN SMITH").

– Nicknames and diminutives: merge with the formal name when descriptions agree (e.g. "Sam" with "Samuel", "Liz" with "Elizabeth", "Bobby" with "Robert").

– Case variants: always merge (e.g. "JOHN" / "John" / "JoHn").

– Spacing / punctuation variants: merge (e.g. "MARY-ANNE" / "MARY ANNE" / "MARYANNE", "MR SMITH" / "MR. SMITH").

– OCR artefacts: merge with the clean form (doubled letters like "JJOOHHNN" = "JOHN", broken hyphenation, stray characters, missing or extra spaces).

– Honorific variants: merge ("DR. CHEN" / "CHEN" / "Dr Chen") when descriptions agree.

WHEN NOT TO MERGE

– Different numbers or ordinals = different people. "GUARD 1" and "GUARD 2" are distinct; "PATRON #1" and "PATRON #2" are distinct; "FIRST THIEF" and "SECOND THIEF" are distinct.

– Formatting variants of the SAME number DO merge: "GUARD 1" / "GUARD #1" / "Guard 1" / "GUARD ONE" are one entity.

– Two characters who share a first name but have clearly different descriptions (different ages, roles, relationships) are distinct – keep them apart even when the surface forms overlap.

– Generic group entities ("BAND", "CROWD", "SOLDIERS", "VAMPIRES") that do not map to a specific named individual stay as their own group; do not fold them into a named character.

– Role labels without a number ("WAITER", "DOCTOR", "REPORTER") are usually one anonymous role and may merge with other unnumbered instances of the same role only when descriptions and scene context clearly point to the same person; otherwise keep separate.

OUTPUT FORMAT (STRICT – a downstream parser depends on it; output ONLY this, no preamble, no markdown, no commentary):

– Output one line per character group, in this EXACT shape:

```
<CANONICAL NAME> <- <alias1>, <alias2>, <alias3>
```

– Left of the " <- " is the canonical name you chose. Right of it is the comma-separated list of EVERY input name in that group.

– The canonical name MUST also appear in its own alias list on the right.

– A character that has no other forms is its own one-item group: write its canonical name on the left and the same single name on the right (e.g. "JOHN <- JOHN").

– Use one line per group. Do not number the lines. Do not add blank lines between groups.

– Use the names EXACTLY as they appear in the input (preserve their original casing and spelling) on the right-hand alias list.

Table 16: **Prompt for resolving coreferent character names (continued).**

Instructions: You read a single scene from a parsed screenplay and extract its canonical events for downstream character-perspective analysis.

Input Information:
SCENE {scene_number}
Heading: {heading}
Location: {location}
Listed characters: {characters}

SCENE SUMMARY (anchor – one line, what this scene is about):
{summary}

SCENE TEXT:
{scene_text}

Extract the canonical events for this scene using the EVENT-block format. If there are no canonical events, output exactly: NO EVENTS

What counts as a canonical event:
– An ATOMIC world-event: one action, one event. A single direction block often describes several atomic events (e.g. "Angier breaks in and reads the journal" = two events: a break-in, then a read of the journal).
– Plot-consequential dialogue beats ARE canonical events: a confession given, a command issued, a public lie told on the record, an oath sworn, an agreement made. Casual conversation is NOT.
– A character speaking only in V.O. (voice-over) or O.S. (off-screen) is narrating, not present. Their V.O. lines are not canonical events.
– A pure transitional moment (title card, montage of stock images with no story action) has NO canonical events. In that case output exactly: NO EVENTS

Atomization rules:
– Group consecutive direction lines describing ONE ongoing action into a single event.
– Split a direction line into multiple events if it describes distinct actions ("A enters and shoots B" = two events).
– Output events in chronological order within the scene.

Table 17: **Prompt for extracting scene events.**

Participants vs witnesses (these are independent dimensions):
– PARTICIPANTS: characters who act in the event or are acted upon by it. Include unconscious victims as participants if the action is done to them.
– WITNESSES: characters who DIRECTLY PERCEIVE the event in real time. Default = the scene's listed characters, MINUS:
– any character whose only presence in the scene is via V.O. or O.S. parenthetical dialogue (they are narrating, not in the room).
– any character the direction marks as unconscious, asleep, blindfolded, looking away, in another room, or otherwise unable to perceive THIS specific event.
– A character can be a participant without being a witness (an unconscious victim) or a witness without being a participant (a bystander).

Character IDs:
– Use the EXACT names from the scene's "Listed characters" line. Do not invent or rename anyone. Do not import characters who are not in that list.
– Never list a V.O.-only narrator as a participant or witness of any event in this scene.

Output format (STRICT – a downstream parser depends on it):
– For each canonical event, write one block in this exact shape:

```
EVENT N
Description: <one sentence, neutral third-person; names from the
listed roster are fine>
Participants: <comma-separated names, or "(none)">
Witnesses: <comma-separated names, or "(none)">
```

– Number events sequentially starting at 1.
– Separate event blocks with a single blank line.
– Each field on its own line; do not break "Description:" across multiple lines.
– Output ONLY the EVENT blocks (or "NO EVENTS"). No preamble, no closing remarks, no markdown, no bullets, no quotes around values.

Table 17: **Prompt for extracting scene events (continued).**

Instructions: You model the first-order belief state of ONE character in ONE scene of a film.

Input Information:
Film: {film_title}

PREVIOUS SCENE:
{prev_scene_context}

CURRENT SCENE:
{current_scene_block}

CANONICAL EVENTS (cite event_ids from here):
{canonical_events}

SCREENPLAY TEXT:
{scene_screenplay}

NEXT SCENE:
{next_scene_context}

TARGET CHARACTER: {character}

{belief_state}

Now decide {character}'s PRESENCE and WITNESSED event_ids for this scene. Output ONLY the two lines specified in your instructions.

You are given: the scene's screenplay text and its canonical events (with stable event_ids); the target character's BELIEF STATE coming into the scene (their current beliefs, active false beliefs, and emotional state) plus the last 1-2 scenes in full; and the next-scene heading for orientation.

Rules:
– Only track what THIS character could know FROM THIS SCENE. Exclude events they did not witness and were not told about.
– A V.O. (voice-over) parenthetical – narrator, phone voice, inner monologue, a letter read aloud – is NARRATING, not physically present. An O.S./O.C. (off-screen / off-camera) parenthetical is the opposite: the character IS in the scene's space, just out of frame (e.g. calling from the next room), and perceives the scene.
– Cite only event_ids that appear in this scene's canonical events, and scene_ids or beliefs that appear in the belief state. Never invent IDs or character names.

YOUR JOB (this prompt): decide the character's PRESENCE this scene and list the canonical events they DIRECTLY witnessed.

Table 18: **Prompt for first-order beliefs, call 1: presence and witnessed events.**

PRESENCE – pick exactly one (a ladder of decreasing access; choose the highest rung that holds):
– on_screen – the scene shows them speaking or acting, or places them physically in its space. A character speaking O.S./O.C. (off-screen / off-camera: in the room but out of frame, e.g. calling from the next room) counts – they perceive the scene. A V.O. (voice-over: narrator, phone voice, inner monologue, a letter read aloud) does NOT – that is narration, not presence.
– offscreen_aware – not in the scene, but a concrete channel makes awareness likely: they were on_screen last scene at the same or an adjacent location and have not left; the previous scene sent them toward this situation; they are stationed within sight/earshot (this one can yield real witnessed events); or the next scene shows them already reacting with no one having told them.
– offscreen_unaware – elsewhere and unaware, BUT the scene bears on them: it changes their world or leaves a belief of theirs now false (dramatic irony). They learn nothing here, yet what they still wrongly believe depends on it.
– absent_unmentioned – a different storyline with no bearing on them at all.

Tie-breakers (walk the ladder top-down, stop at the first rung you can justify):
– on_screen vs offscreen_aware: if you cannot place them in the scene's space, drop to offscreen_aware.
– offscreen_aware vs offscreen_unaware: pick aware only if you can NAME the concrete channel; otherwise unaware.
– offscreen_unaware vs absent_unmentioned: pick unaware if the scene touches their situation or beliefs; absent only if it is truly unrelated.

WITNESSED: list only event_ids the character DIRECTLY perceives (sees/hears/feels firsthand). Second-hand information is NOT witnessed (it is handled as `told` in the next step). Being on_screen does not force entries here – someone present but asleep, unconscious, or away from the action witnesses nothing. Normally empty when offscreen or absent.

Output ONLY these two lines, nothing else:

```
Presence: <on_screen|offscreen_aware|offscreen_unaware|
absent_unmentioned>
Witnessed event IDs: <comma-separated event_ids, or "(none)">
```

Table 18: **Prompt for first-order beliefs, call 1: presence and witnessed events (continued).**

Instructions: You model the first-order belief state of ONE character in ONE scene of a film.

Input Information:
Film: {film_title}

PREVIOUS SCENE:
{prev_scene_context}

CURRENT SCENE:
{current_scene_block}

CANONICAL EVENTS (cite event_ids from here):
{canonical_events}

SCREENPLAY TEXT:
{scene_screenplay}

NEXT SCENE:
{next_scene_context}

TARGET CHARACTER: {character}

{belief_state}

CALL 1 RESULT (presence + witnessed_event_ids for this scene – treat as ground truth):
{call1_text}

Now produce TOLD, INFERRED, FALSE BELIEFS HELD, and REVISIONS for {character} in this scene, using the exact four-section format specified in your instructions.

You are given: the scene's screenplay text and its canonical events (with stable event_ids); the target character's BELIEF STATE coming into the scene (their current beliefs, active false beliefs, and emotional state) plus the last 1-2 scenes in full; and the next-scene heading for orientation.

Rules:
– Only track what THIS character could know FROM THIS SCENE. Exclude events they did not witness and were not told about.
– A V.O. (voice-over) parenthetical – narrator, phone voice, inner monologue, a letter read aloud – is NARRATING, not physically present. An O.S./O.C. (off-screen / off-camera) parenthetical is the opposite: the character IS in the scene's space, just out of frame (e.g. calling from the next room), and perceives the scene.
– Cite only event_ids that appear in this scene's canonical events, and scene_ids or beliefs that appear in the belief state. Never invent IDs or character names.

YOUR JOB (this prompt): given Call 1's presence + witnessed result (treat it as ground truth), extract what the character was TOLD this scene, what they INFERRED, the FALSE BELIEFS they currently hold, and any REVISIONS to prior beliefs.

Table 19: **Prompt for first-order beliefs, call 2: told, inferred, false, and revised beliefs.**

TOLD – propositions communicated to the character via dialogue this scene. Extract the proposition, not the wording ("Smoke says the deal is off" → "the deal is off"). Mark believed yes/no. A character who is not present cannot be told anything here.
INFERRED – NEW claims the character draws from witnessed + told + their current beliefs. Each cites a basis: event_ids from this scene and/or scene_ids. Do NOT re-emit beliefs already listed in the belief state.
FALSE BELIEFS HELD – beliefs the character holds that contradict the canonical events. Carry forward the belief state's active false beliefs unless this scene corrects them; add any new ones introduced by deception or mistaken inference. Write each false belief in FULL TEXT here (do not use a [Bk]/[Fk] tag in this section).
REVISIONS – when this scene corrects a prior belief: in `Prior`, cite that belief's [Bk] or [Fk] tag from the belief state (e.g. [B3]); write it out verbatim only if it has no tag there. Then give what it is revised to and the triggering event_ids.

Output ONLY these four sections, in this order. Each section's body is a list of items or the literal "(none)". Separate items with a blank line.

```
TOLD
- Proposition: <claim>
  Source: <CHAR>
  Believed: <yes|no>

INFERRED
- Claim: <claim>
  Basis: <comma-separated event_ids and/or scene_ids>
  Confidence: <high|medium|low>

FALSE BELIEFS HELD
- <belief>

REVISIONS
- Prior: <[Bk]/[Fk] tag from the belief state, e.g. [B3] -- or
verbatim text if untagged>
  Revised to: <new belief>
  Trigger: <comma-separated event_ids>
```

No preamble, no closing remarks, no markdown headers.

Table 19: **Prompt for first-order beliefs, call 2: told, inferred, false, and revised beliefs (continued).**

Instructions: You model character c's META-BELIEFS – what c believes about OTHER characters' mental states – for ONE scene of a film. Second-order only (c about d): what c thinks d believes. Never third order.

Input Information:
Film: {film_title}
SCENE ID: {scene_id}

PREVIOUS SCENE:
{prev_scene_context}

CURRENT SCENE:
{current_scene_block}

CANONICAL EVENTS (prefer these event_ids for Basis; use this scene's id s_... for inference/ absence beliefs with no discrete event):
{canonical_events}

SCREENPLAY TEXT:
{scene_screenplay}

NEXT SCENE:
{next_scene_context}

MODELING CHARACTER (c): {character}
TARGET GROUP THIS CALL (model ONLY these {character} → d): {target_set}

C'S FIRST-ORDER UPDATE FOR THIS SCENE:
{c_first_order_update}

{meta_state}

Now emit {character}'s NEW or CHANGED meta-beliefs about the targets in the group above (any of the three types; the Claim is the bare proposition d believes). Skip beliefs already held unchanged. OMIT targets with nothing new – do not enumerate them. If no target has anything new, output the single word NONE. Use the exact per-target block format from your instructions.

You are given: the scene's screenplay text and canonical events (with stable event_ids); c's first-order update for THIS scene (what c witnessed, was told, inferred); c's meta-beliefs held so far plus the last 1-2 scenes in full; and a SMALL GROUP of target characters – the only characters you may model in THIS call.

Table 20: **Prompt for meta-beliefs.**

For each target d in the group, emit c's NEW or CHANGED meta-beliefs about d's mind – of any of the three types below. Model ONLY the targets in this call's group; never emit about a character outside it.

THE THREE TYPES (tag every item with exactly one; they carry equal weight – do not default to any single type):

– first_order_about_d – c thinks d believes a proposition about the WORLD (not about c, not about a specific third character). This is the MOST COMMON type: what d knows or believes about events, facts, and situations. E.g. "the deal is off"; or via d_absence, c thinks d does NOT know something that happened.

– about_self – c thinks d believes a proposition ABOUT c personally. E.g. "c is the new pastor". (Relevant to deception: if c lied to d, or someone lied about c in d's hearing, c gains an about_self belief that d holds a possibly-false belief about c.)

– about_third_party – c thinks d believes a proposition about a THIRD named character e (e is different from both c and d). Name e in the Claim; e must already appear in the canonical events or prior history.

CLAIM PHRASING (strict): the Claim is the BARE PROPOSITION that d believes – nothing more. The About field already names the believer, so NEVER write "d believes/knows/thinks ..." inside the Claim. Write the proposition itself ("the deal is off", "c is the new pastor", "e stole the money"). Encode how strongly d holds it via Source and Confidence, not with a verb in the claim. If the proposition is ITSELF about someone's belief (a belief about a belief), DROP it – this extractor is second-order only.

VALID SOURCES (pick exactly one per item):

– shared_witnessing – c and d both witnessed event e; c gains "d knows e".

– observed_d_told – c saw someone tell d a proposition.

– observed_d_behavior – d acted in a way implying a belief; c infers it (lower confidence).

– c_told_d – c directly communicated something to d.

– d_absence – d was not present and has no plausible channel; c believes d does NOT know (reserve for gaps that matter, not routine).

– shared_prior_context – mutual background knowledge (use sparingly).

Table 20: **Prompt for meta-beliefs (continued).**

GROUNDING (every item MUST cite a Basis):
– PREFER a canonical event_id from THIS scene (e.g. e_017_3) – the specific event that supports the claim.
– When the belief comes from dialogue or stage direction with no discrete canonical event – common for observed_d_behavior and shared_prior_context – cite THIS scene's id (e.g. s_017).
– For d_absence (c believes d does NOT know something), cite the PRIOR scene_id where the missed event occurred.
No valid basis, no item – drop it rather than invent an id. Never fabricate event_ids, scene_ids, or character names.

NOVELTY: emit a meta-belief when it is NEW this scene, when this scene REVISES one c already holds (a CHANGED belief – cite the prior scene_id plus this scene's trigger), or when this scene gives materially stronger evidence for a previously shaky belief. Do NOT re-state beliefs already in "meta-beliefs held so far" that are unchanged.

DO NOT EMIT (these are not meta-beliefs):
– trivially-true standing facts, static traits, or roles ("d is a bartender", "d is c's employer").
– public world-facts every observer already shares, with no divergent knowledge ("the band is playing").
– bare narrative summaries or predictions no named d actually holds ("the meeting ended without incident").
– perceptions or imperatives ("d looks at c", "d should leave").
– the SAME observation emitted twice under two types – pick the single best type.

A meta-belief may be wrong: c can believe d knows something d does not. That is valid – capture what c believes, not the ground truth.

OUTPUT FORMAT (STRICT – a downstream parser depends on it). Emit one block per meta-belief. OMIT targets you have nothing new for – do not enumerate them. If NO target in the group has anything new or changed this scene, output the single word NONE and nothing else.

```
- About: <D -- a target in this group>
  Claim: <the bare proposition d believes>
  Type: <first_order_about_d|about_self|about_third_party>
  Source: <shared_witnessing|observed_d_told|observed_d_behavior|
c_told_d|d_absence|shared_prior_context>
  Basis: <comma-separated event_ids and/or scene_ids>
  Confidence: <high|medium|low>
```

Table 20: **Prompt for meta-beliefs (continued).**

WORKED EXAMPLE (neutral, not from this film). Scene s_042: c=Rosa and Dev both watch the foreman announce the factory will close Friday; Rosa then quietly tells Dev she has already lined up new buyers. Priya, a partner, is not present. Group: Dev, Priya. Output:

```
- About: Dev
  Claim: the factory will close on Friday
  Type: first_order_about_d
  Source: shared_witnessing
  Basis: e_042_3
  Confidence: high
- About: Dev
  Claim: Rosa has already lined up new buyers
  Type: about_self
  Source: c_told_d
  Basis: e_042_7
  Confidence: high
- About: Priya
  Claim: the factory will close on Friday
  Type: first_order_about_d
  Source: d_absence
  Basis: s_042
  Confidence: medium
```

(The plain world-belief is the common case; about_self and about_third_party are the minority. The last item shows d_absence for an absent partner, grounded on the scene id – use it only when the knowledge gap matters.)

Output ONLY the per-target blocks (or NONE). No preamble, no closing remarks, no markdown headers.

Table 20: **Prompt for meta-beliefs (continued).**

Type	Specification injected into the question-writing prompt
Q1 Knowledge split	Cohort knowledge split — at one moment several people are in different epistemic states. What does each take to be the case?
Q2 Realization	Realization moment — X's understanding changes at a locatable point in the film. The answer must say when; the question must not say what.
Q3 False belief	In-scene false belief at a threshold — right before an act, X holds a belief the film has shown to be wrong. Ask what X takes to be the case. Do NOT signal that X is mistaken; noticing the gap is the task.
Q4 Knowledge source	Source attribution — a fact reaches people by different routes (c_told_d, observed_d_behavior, shared_witnessing, d_absence), and one scene often carries more than one route. Ask how someone comes to know it, open. Never offer the routes as alternatives to pick between.
Q5 First-order belief	First-order ToM — X holds a model of what Y believes. Ask what X takes Y to believe.
Q6 Second-order belief	Higher-order ToM — X holds a model of what Y believes about Z.
Q7 Accepting a claim	Deception detection — X is told things; believed_by_c marks which X accepts and which X rejects, and BOTH happen across this dataset. Ask what X makes of the claim. Never phrase it as if either outcome were a given, and do not ask the film to adjudicate the claim's truth in the question — the answer settles that.
Q8 Order of discovery	Temporal ordering of learning — X's discoveries, in order.
N1 Epistemic arc	Epistemic-arc summary — what X comes to know across the film, in order.
N2 Trust evolution	Relationship/trust evolution — how X's model of Y changes across the film.
N3 Causal chain	Causal chain — what chain of events leads X to believe Z.

Table 21: **The per-type specification.** One line per question type, inserted verbatim into the question-writing prompt after the system prompt. Each names the epistemic structure the question must require and stops; it does not prescribe the question's wording. The field names it mentions (`c_told_d`, `believed_by_c`) are the belief record's own; the question writer is told to keep them out of the question.

Instructions: You write questions for a film-QA dataset. You see structured belief annotations; the test-taker sees ONLY the film. Your question is good if a viewer who watched carefully answers it and a viewer who watched carelessly gets it wrong. It is bad if it can be answered from the question itself, from the question's shape, or from genre convention.

Input Information:
QUESTION_TYPE: {q_type} – {type_specification}

ANSWER CONTRACT (applies to every question of every type – do not restate it in the question text):
– Every claim in answer_outline names the on-screen basis it rests on, in source_path.
– Where a type turns on HOW something was learned (told / observed / witnessed together / inferred from an absence), the answer says which. The question does not.
– Where a type turns on WHEN, the answer locates the moment. The question does not.
– Where the holder is wrong, the answer says so. The question does not.

EXEMPLARS (a different film; imitate the shape, not the content):
{two_example_triples}

CONTEXT PACK:
{context_set_json}

READING THE PACK. Four parts, and the third is the one to read first.
– `basis` – the belief records the question is built from. `claim` is what is believed; `source` is how it was acquired; `basis` lists the canonical events it rests on.
– `scenes` – pass-1 canonical events with `participants` and `witnesses`. This is the ground truth for who could possibly have known something. If your question implies someone knew a thing, they must be in one of these lists.
– `basis[].target_state` – WHAT THE OTHER PERSON ACTUALLY BELIEVES, from a separate extraction pass, at this same scene. A record says "X believes Y believes P"; `target_state` says what Y believes in their own right. **Compare them.** Where X's model of Y diverges from Y's own state – especially where `target_state` carries `false_beliefs_held` – that gap is the question. Where they agree, X is simply right about Y and there is usually nothing to ask.
– `name_budget` / `named_later` / `forbidden_names` / `notes` – see NAMING below.

Table 22: **Prompt for writing the question (Stage 1).**

REFUSE A THIN PACK. Before writing, ask: is there a specific wrong answer a careful person might give here? If not – the belief is uncontested, X's model of Y matches Y exactly, the belief rests on a single unremarkable action, or the holder is someone the film never picks out – set verdict="thin", say why in thin_reason, leave question empty, and stop. A refused pack costs one call. A thin question costs a rubric, a grading pass, and a benchmark number that means nothing.

THE ANCHOR – point at ONE moment, do not narrate it.

– At most 12 words before your first question word, or 18 if you need a disambiguator (below).

– Name a PLACE, an OCCASION or an OBJECT ("in the utility hallway", "on the publishers' conference call", "at the beach overlook"). Never report what anyone does, decides, learns, reveals, wins, accepts or refuses there.

– Anchor diegetically: never a scene number, event id, or annotation vocabulary.

– The test-taker's job includes FINDING the moment. A synopsis does that job for them, and smuggles the answer's setup in as premise.

– **CHECK `anchor_location_rivals` FIRST.** It counts other scenes in this film at the same location with this holder present. You cannot work this out yourself – you see three scenes of about a hundred and twenty. If it is 0, the place alone is enough. If it is 1 or more, naming the place is NOT enough and your question has several possible answers: add ONE distinguishing detail that is not an outcome – a time ("months after the injury"), another person present ("with Art"), or an object ("over the fixed poker table"). Those locate the moment without saying what anyone believes, which is the whole point. Never disambiguate with what happens there.

– The location string can also disagree with what the film shows (an annotation labelled a scene "CORNER SUITE 2" whose own summary opens "Charlie enters Sterling's office"). Prefer what the SUMMARY shows a viewer over the location label.

TOPIC, NEVER VALUE. Name what the belief is ABOUT. Never state what it says, whether it is true, or that it is contested.

– Yes: "what does Bosco make of Monk's attitude to the new title"

– No: "...and what does Monk's reaction show is actually true" – announces the gap, which IS the question.

– No: "when does Monk first realize Cliff will not contribute" – the content of the realization is the answer; only its timing is left.

– No: "who is still reading it as just a routine drill" – hands over the mistake.

– Test yourself: strike every word of your question that the gold answer also needs. If the gold is now mostly written, rewrite the question.

Table 22: **Prompt for writing the question (Stage 1) (continued).**

ONE OPEN INTERROGATIVE. A question word already forbids a yes/no answer; you do not need to defend against it.

– No "A, or B?" alternatives. They are coin flips with a prior: 57% of this dataset's source-attribution answers read "inferred from behaviour", so a model that always guesses that scores without watching anything.

– No ", and what supports that reading?" or "...and what gives it away?" tails. Every answer must cite its on-screen basis – that is a standing requirement of the answer, not something to announce in each question.

VARY THE FRAME. If every question of a type opens the same way, the type is the answer key. Take the exemplars' level of specificity, not their sentence shape.

NAMING. Freely use name_budget. You may use named_later – the viewer watches the whole film – unless the name is itself the reveal. NEVER use forbidden_names: the film never says them aloud, so no viewer could connect one to a person. The scene text names people the film does not; that is the annotation talking. Follow the lists, not the prose.

ANSWER OUTLINE: 2 to 4 claims, no more. Each is one assertion about who believes what, written as a viewer would see it, never as the annotation says it. Do not write "the belief annotation marks X"; write what happens, and put every annotation reference (event ids, scene ids, basis fields) in source_path.

DISTRACTOR: required and concrete. Write the specific wrong answer you expect – the inversion, the wrong holder, the surface event mistaken for the belief. Not "a viewer might be confused"; a sentence someone would actually write.

Return ONLY the JSON object.

Table 22: **Prompt for writing the question (Stage 1) (continued).**

Instructions: You write the gold answer for a film-QA dataset, and the CLAIM SET it will be graded by. Both come from the ANSWER_OUTLINE; you may not add content to either.

Input Information:
QUESTION_TYPE: {q_type}
QUESTION: {question}

ANSWER_OUTLINE:
{outline_claims}

NAMING: {name_lists}

THE CONTRACT. Every claim you return must be plainly asserted by your prose, and every substantive assertion in your prose must be backed by a claim you return. The claim set IS the grading rubric: an answer is scored by which of these claims it asserts and which it contradicts. A claim your prose only implies will fail that test – assert it or drop it.

PROSE. 60-140 words, as a careful viewer would give it – natural register: normal capitalisation (Charlie Cale, never CHARLIE CALE), pronouns are fine once a person is established, no filler ("In other words..."). Hedge claims the outline hedges; never hedge a confident claim into vagueness. Name evidence moments approximately ("just after the argument on the stairs"), never scene ids, event ids, or timestamps in seconds. Use the question's naming rules: name_budget and named_later freely, forbidden_names never – describe those people instead.

Table 23: **Prompt for writing the reference answer and rubric (Stage 2).**

CLAIMS. Rewrite each outline claim you keep as one self-contained, judgeable proposition, $\leq$25 words: name the people (no pronouns), one assertion each, no annotation vocabulary, non-overlapping. Keep the outline's meaning exactly – do not strengthen, weaken, or merge. A grader reads each claim alone against a candidate answer and must be able to say asserted / contradicted / omitted.

– ATOMIC: one assertion means one. Never join assertions with "while", "because", "and then". Split a compound outline claim into several claims sharing its outline_index – a grader must never have to decide which half of a claim an answer satisfied.

– NO LOCATORS: the claim asserts content, not place or time. "On Arthur's phone call, Monk learns of the offer" penalises an answer that has Monk learning it but names the moment differently – write "Monk learns that a publisher offered $750,000". Where and when belong to the question and the prose.

– Names in claims are written normally (Charlie Cale), never in screenplay capitals (CHARLIE CALE).

ROLE – what the QUESTION requires, not what the outline contains. For each claim:

– "core": a complete, correct answer to THE QUESTION must assert it. Ask: could an answer omit this and still fully answer what was asked? If yes, it is not core.

– "context": premise, setup, or supporting fact – restating the question's own scenario, naming who spoke, background the question did not ask to be traced. A correct answer may omit it; only contradicting it is an error.

Answers are graded on recall over CORE claims alone. A padded core set punishes every correct answer for the padding; when unsure, "context". At least two claims should be core – fewer means the outline did not carry the question's content, which is worth saying in dropped_why rather than inflating a context claim.

DROPPING. If an outline claim cannot be honestly asserted in the prose – too weakly grounded, or it would not survive a viewer's scrutiny – drop it: list its index in dropped and say why in dropped_why. A dropped claim costs nothing; a claim the gold itself does not assert poisons every grade computed with it. Never pad the prose to cover a claim.

Return ONLY the JSON object.

Table 23: **Prompt for writing the reference answer and rubric (Stage 2) (continued).**

Instructions: You verify the claims of a film-QA gold answer against the film's SCREENPLAY. The claims were derived from an automatic belief annotation that is sometimes wrong; the script is the ground truth here. Judge ONLY from the text shown. Film knowledge, plausibility, and narrative sense are not evidence.

Input Information:
FILM: {film_id}

*STORY CONTEXT (every scene of the film; heading + short neutral summary, film order; * = shown in full below)*:
{one_line_per_scene}

QUESTION: {question}
HOLDER (the character the question is about): {center_character}

CLAIMS:
{numbered_claims}

SCRIPT (full text; scenes in film order; the question's anchor is {anchor_scene_id}):
{scenes_shown_in_full}

For each numbered CLAIM return one label:
– "supported" – the script shown establishes it; a careful reader must conclude it.
– "unsupported" – the script shown does not establish it (silence, absence, or an inference the text does not force).
– "contradicted" – the script shown says otherwise.
Default to "unsupported" when unsure: a wrong claim corrupts every answer graded against it, while a dropped claim costs one point of coverage.

PRESENCE. A claim that puts a person at a moment (witnesses, hears, sees, is told, learns) needs that person physically there in the text. A character who is only mentioned is not present. A character on the other end of a phone call or an INTERCUT conversation perceives only what is said on the line – never gestures, looks, or actions in the other room. Someone asleep, unconscious, out of earshot, or elsewhere perceives nothing. The cast lists shown come from a parser and may include people who are merely mentioned; the text decides.

Table 24: **Prompt for verifying claims against the screenplay (Stage 3).**

BELIEF CLAIMS – the rules that matter:

– "X believes P" is supported when X asserts P, acts on P, endorses or goes along with P, or is told P and shows no doubt. Do not require the proposition to be spoken verbatim; a response that builds on P shows belief in P. X being absent when not-P happened is NOT belief in P.

– "X knows / understands / learns Q": supported when X witnesses Q, is told Q, or is present when Q becomes evident – a realization by witnessing needs no statement. Contradicted when the film shows X acting on not-Q or being kept from Q (check the WHOLE STORY CONTEXT, not just the anchor).

– "X falsely believes P" / "X is wrong" / "X's belief is mistaken" has two halves. The belief half follows the rules above. The falsity half is judged against the WHOLE film: it is supported only when some scene establishes not-P, and it is CONTRADICTED when the film shows P to be true – a belief the story treats as delusion early and vindicates late is a true belief. If the belief half is supported and the falsity half is merely not established, label the claim "supported" and put the claim REWRITTEN WITHOUT the qualifier in "rewrite" (same meaning otherwise, $\leq$25 words, names not pronouns). A claim that consists ONLY of the qualifier ("X's belief is wrong") is contradicted when P is true and unsupported when nothing settles it.

– "X does not know Q": supported only if Q happened where X was absent and no scene in the film shows X told or learning it; ignorance by silence alone is unsupported.

– "X takes / thinks / believes Y believes P" is supported only with an observed channel: X sees or hears Y assert or act on P; X tells Y P; X is present when Y is told P; or X and Y both witness the event P describes. In every case the text must let X identify what P is about – overhearing one side of a phone call to an unnamed party is not a channel. X's own view of P projected onto Y is unsupported. If Y visibly does not hold P in X's presence, it is contradicted.

– "X accepts / believes / rejects what Y said" needs a visible response from X – words, action, or going along with it. Silence is unsupported. The tell itself ("Y tells X P") is supported if the text shows it with X present.

– "X learns P by witnessing" / "by being told" / "by inference": the route must be the one in the text.

– "X revises / realizes / changes their mind": the prior belief must be visible in the same story period. A belief from a flashback or a different year does not carry into another period; a "revision" whose prior lives on the other side of a time jump is unsupported.

– "A happens before B": judge from scene order; no belief reasoning needed.

STORY CONTEXT is the WHOLE film: one line per scene, in film order, with the heading and a short neutral summary from an independent pass; the scenes shown in full below are marked *. Read all of it before judging. Use it for what happened anywhere in the film, who was present where, the order of events, which story period a scene belongs to, whether a character ever learns something, and whether a belief is true or false in the end. Never use it to establish what anyone believes or thinks – only the full text can – and never use it to contradict a belief a character visibly holds in the full text.

Table 24: **Prompt for verifying claims against the screenplay (Stage 3) (continued).**

AFTER THE VERDICTS – the surviving set. Take the claims you labelled "supported" (with any rewrites applied). Then:

– "answers_question": "yes" if a complete answer built from the surviving CORE claims alone still fully answers the QUESTION as asked; "no" if the dropped claims were what the question was about (e.g. the question asks what a person works out, and every surviving claim is about someone else).

– "revised_gold": if you dropped or rewrote ANY claim, write the gold answer again as 60-140 words of plain prose that asserts every surviving claim, each recognisably, and asserts NOTHING beyond them – no dropped content, no new facts, names in normal case, no scene ids. Empty string if nothing changed.

– "revised_question": only when answers_question is "no" AND at least two surviving claims are core: write ONE replacement question that those surviving core claims answer completely, following these rules exactly – one open interrogative (what / how / when / who), never a yes/no or an "A or B"; open with a short diegetic anchor (a place, occasion or object, at most 12 words, never an outcome and never a scene id); ask about the TOPIC of the belief, never state its value or that it is false; no ", and what supports that" tail; use only character names that the original question or the surviving claims already use; the question must not be answerable from its own wording. If the surviving claims are only events with no belief, knowledge or perception in them, write the question anyway and type it honestly below. Empty string otherwise. Then also fill revised_gold for the revised question.

– "revised_type": the type the revised question tests, one of: Q1 cohort knowledge split (what each of several people takes to be the case at one moment); Q2 realization moment (when someone's understanding changes); Q3 in-scene false belief (what someone takes to be the case right before acting); Q4 source attribution (how a fact reaches someone); Q5 first-order ToM (what X takes Y to believe); Q6 higher-order ToM (what X takes Y to believe about Z); Q7 deception detection (what X makes of what X is told); Q8 ordered discoveries; N1 epistemic arc across the film; N2 how X's model of Y changes; N3 causal chain to a belief. Empty string when there is no revised question.

– "revision_note": one sentence on why the question still holds, or what it now asks instead. Empty string if nothing changed.

Return ONLY the JSON object: one verdict per claim id ("rewrite" empty unless the qualifier rule applies, "why" one short sentence citing the scene), then answers_question, revised_gold, revised_question, revised_type, revision_note.

Table 24: **Prompt for verifying claims against the screenplay (Stage 3) (continued).**

Instructions: You grade an ANSWER against the CLAIMS of a film-QA question.

Input Information:
CLAIMS:
{numbered_claims}

ANSWER:
{answer}

For each numbered claim, exactly one label:

– "asserted" – the answer states the claim's content, in any wording. Hedged but committed ("she seems not to buy it") still asserts.

– "contradicted" – the answer states something incompatible with the claim: the opposite polarity, the wrong holder, the wrong source, the inverted direction. Merely discussing the topic is not this.

– "omitted" – neither: the answer does not address the claim's content, or addresses it too vaguely to commit either way. An answer that declines the question omits every claim.

Judge each claim independently, only against the ANSWER's text. The answer's extra content is not graded here and must not influence any verdict.
Return ONLY the JSON object, one verdict per claim id.

Table 25: **Prompt for the judge, which grades an answer against its rubric.**